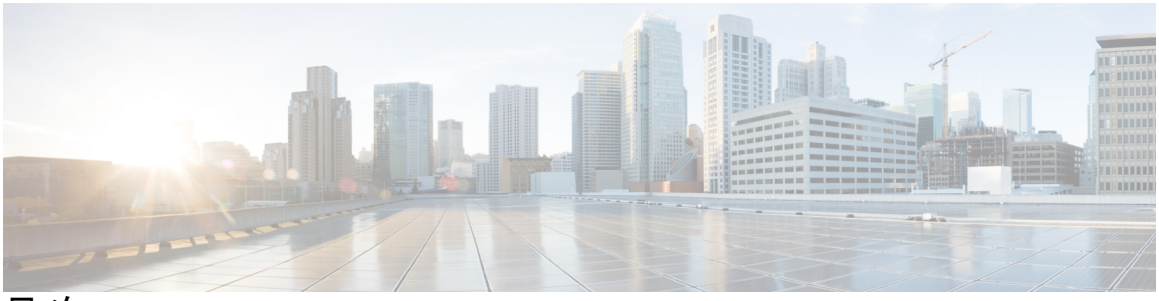
Cisco Secure 動的属性コネクタ 2.2 構成ガイド

初版：2023 年 7 月 7 日

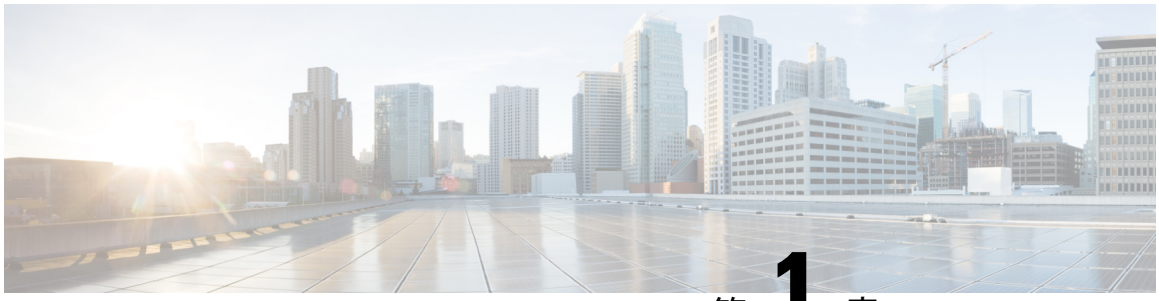
最終更新：2023 年 7 月 17 日

シスコシステムズ合同会社
〒107-6227 東京都港区赤坂9-7-1 ミッドタウン・タワー
http://www.cisco.com/jp
お問い合わせ先：シスコ コンタクトセンター
0120-092-255 （フリーコール、携帯・PHS含む）
電話受付時間：平日 10:00～12:00、13:00～17:00
http://www.cisco.com/jp/go/contactcenter/



目 次

【注意】シスコ製品をご使用になる前に、安全上の注意（ **www.cisco.com/jp/go/safety_warning/** ）をご確認ください。本書は、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。また、契約等の記述については、弊社販売パートナー、または、弊社担当者にご確認ください。



第 1 章

Cisco 動的属性コネクタについて

Cisco Secure 動的属性コネクタ では、クラウドプロバイダーからデータ（ネットワークや IP アドレスなど）を収集し、それを Ciso Secure Firewall Management Center（management center）に送信して、アクセス制御ルールで使用できるようにします。

次のトピックでは、動的属性コネクタ に関する背景について説明します。

Cisco Secure 動的属性コネクタについて

Cisco Secure 動的属性コネクタ により、さまざまなクラウドサービス プラットフォームのサービスタグとカテゴリを Secure Firewall Management Center（ management center）アクセス制御で使用できます。

サポートされるコネクタ

現在、次をサポートしています。

表 *1 : Cisco Secure* 動的属性コネクタ バージョンおよびプラットフォーム でサポートされているコネクタのリスト

CSDAC バージョン/プラットフォーム	AWS	GitHub	Google クラウド	Azure	Azure サービスタグ	Microsoft Office 365	vCenter
バージョン 1.1（オンプレミス）	対応	×	×	対応	対応	対応	対応
バージョン 2.0（オンプレミス）	対応	対応	対応	対応	対応	対応	対応
バージョン 2.2（オンプレミス）	対応	対応	対応	対応	対応	対応	対応

コネクタの詳細は次のとおりです。

- Amazon Web Services（AWS）

詳細については、Amazon ドキュメントサイトの「AWS リソースのタグ付け」などのリソースを参照してください。

- GitHub

  詳細については、GitHub コネクタの作成 （25 ページ）を参照してください。

- Google クラウド

  詳細については、 Google Cloud ドキュメントの「環境設定」を参照してください。

- Microsoft Azure

  詳細については、Azure ドキュメントサイトのこのページを参照してください。

- Microsoft Azure サービスタグ

  詳細については、Microsoft TechNet の「仮想ネットワークサービスタグ」などのリソースを参照してください。

- Office 365 の IP アドレス

  詳細については、docs.microsoft.com の「Office 365 URL および IP アドレス範囲」を参照してください。

- vCenter と NSX-T によって管理される VMware のカテゴリとタグ

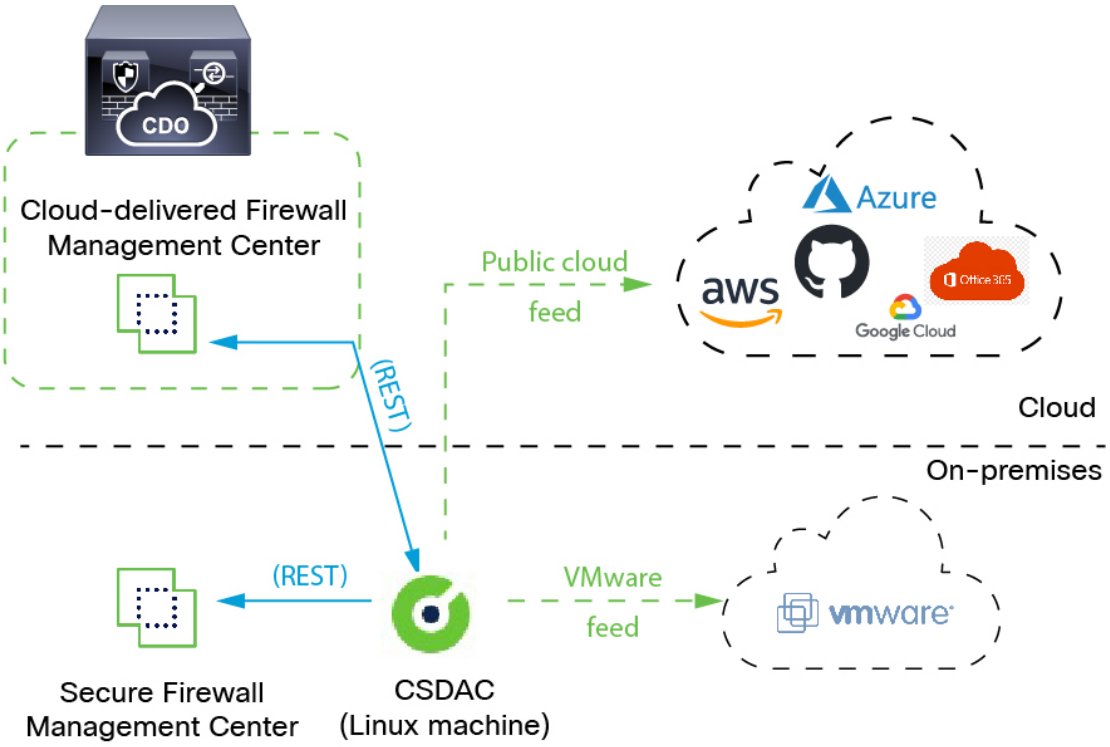
  詳細については、VMware ドキュメントサイトの「vSphere タグと属性」などのリソースを参照してください。

機能の仕組み

ワークロードの動的な性質とIPアドレスの重複の必然性により、IPアドレスなどのネットワーク構造は、仮想、クラウド、およびコンテナ環境では信頼できません。お客様は、IPアドレスやVLANが変更されてもファイアウォールポリシーが持続するように、VM名やセキュリティグループなどの非ネットワーク構造に基づいてポリシールールを定義する必要があります。

これらのタグと属性は、Ubuntu、CentOS、または Red Hat Enterprise Linux 仮想マシンで実行されている 動的属性コネクタ Docker コンテナを使用して収集できます。Ansible コレクションを使用して、Ubuntu ホストに 動的属性コネクタ をインストールします。

次の図は、システムが高レベルでどのように機能するかを示しています。



- サポートされている Linux 仮想マシンに 動的属性コネクタ をインストールします。

  詳細については、サポートされているオペレーティングシステムとサードパーティソフトウェア （5 ページ）を参照してください。

- システムは、特定のパブリック クラウド プロバイダーをサポートします。

  このトピックでは、サポートされているコネクタ（これらのプロバイダーへの接続）について説明します。

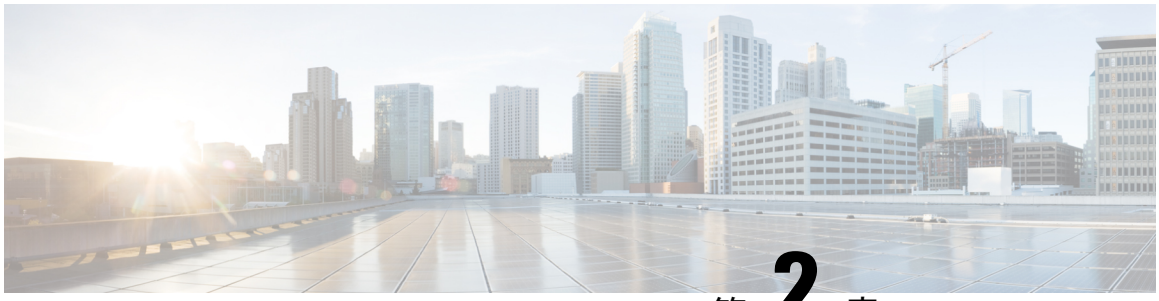
- 動的属性コネクタ によって定義されたアダプタは、これらの動的属性フィルタをダイナミックオブジェクトとして受け取り、アクセス制御ルールで使用できるようにします。

  次のタイプのアダプタを作成できます。

  - オンプレミスの Management Center デバイスの場合、オンプレミス *Firewall Management Center*。

    このタイプの Management Center デバイスは、Cisco Defense Orchestrator（CDO）によって管理されるか、スタンドアロンである可能性があります。

  - CDO が管理するデバイスの場合、クラウド提供型 *Firewall Management Center* 。



第 2 章

Cisco Secure 動的属性コネクタのインストールおよびアップグレード

この章では、サポートされているすべてのオペレーティングシステムに Cisco Secure 動的属性コネクタをインストールする方法について説明します。

サポートされているオペレーティングシステムとサードパーティソフトウェア

動的属性コネクタ の前提条件は次のとおりです。

- Ubuntu 18.04 ～ 22.04.2
- CentOS 7 Linux または 8
- Red Hat Enterprise Linux（RHEL）7 または 8
- Python 3.6.x 以降
- Ansible 2.9 以降

すべてのオペレーティングシステムの最小要件：

- 4 個の CPU
- 8 GB RAM
- 新規インストールの場合は、100GB の空きディスク容量

vCenter 属性を使用する場合は、次も必要です。

- vCenter 6.7
- 仮想マシンに、VMware ツールがインストールされている必要があります。

仮想マシンのサイジング

次のように仮想マシンのサイズを設定することを推奨します。

- 50 個のコネクタ（コネクタごとに 5 つのフィルタと 20,000 ワークロードを想定）、4 つの CPU、8GB RAM、100 GB の空きディスク容量
- 125 個のコネクタ（コネクタごとに 5 つのフィルタと 50,000 ワークロードを想定）、8 つの CPU、16GB RAM、100 GB の空きディスク容量

（注） 仮想マシンのサイズを適切に設定しないと、動的属性コネクタに障害が発生したり、起動しなかったりする可能性があります。

前提条件ソフトウェアのインストール

始める前に

物理的または仮想的な設定があり、システムが オンプレミス Firewall Management Center または クラウド提供型 Firewall Management Center と通信できることを確認してください。

ステップ 1 （オプション）テキストエディタを使用して `/etc/environment` を編集し、次の変数をエクスポートして、Ubuntu マシンがインターネットプロキシの背後にある場合にインターネットと通信できるようにします。

変数	値
export http_proxy	HTTP プロキシで使用します。 *user*:*pass*@*host-or-ip*:*port*
export https_proxy	HTTPSプロキシでこれを使用します。 *user*:*pass*@*host-or-ip*:*port*
export no_proxy	プロキシ構成を削除します。 **export no_proxy="localhost,127.0.0.1"**

例：

認証なしの HTTP プロキシ：

```
vi /etc/environment
export http_proxy="myproxy.example.com:8181"
```

認証付き HTTPS プロキシ：

```
vi /etc/environment
export https_proxy="ben.smith:bens-password@myproxy.example.com:8181"
```

ステップ 2 別のコマンドウィンドウを使用して設定を確認します。

```
env grep | proxy
```

結果の例：

```
http_proxy=myproxy.example.com:8181
```

ステップ 3 次のいずれかのセクションに進みます。

関連トピック

前提条件ソフトウェアのインストール：Ubuntu （10 ページ）
前提条件ソフトウェアのインストール：CentOS （7 ページ）
前提条件ソフトウェアのインストール：RHEL （8 ページ）

前提条件ソフトウェアのインストール：CentOS

始める前に

次のことをすべて行います。

- システムがサポートされているオペレーティングシステムとサードパーティソフトウェア（5 ページ） で説明した前提条件を満たしていることを確認します。
- （オプション）動的属性コネクタへのプロキシアクセスが必要な場合は、前提条件ソフトウェアのインストール （6 ページ）を参照してください。

ステップ 1 Docker がインストールされていないことを確認し、インストールされている場合はアンインストールします。

```
docker --version
```

Docker がインストールされている場合は、Ubuntu での Docker エンジンのアンインストールの説明に従ってアンインストールします。

ステップ 2 リポジトリを更新およびアップグレードします。

CentOS 7：

```
sudo yum -y update && sudo yum -y upgrade
```

ステップ 3 epel リポジトリをインストールします。

CentOS 7：

```
sudo yum -y install epel-release
```

ステップ 4 （CentOS 7 のみ。）Python3 をインストールします。

```
sudo yum install -y python3 libselinux-python3
```

ステップ 5 Ansible をインストールします。

CentOS 7 および CentOS 8：

```
sudo yum install -y ansible
```

ステップ 6 Ansible のバージョンが 2.9 以降であることを確認します。

CentOS 7：

```
ansible --version
  ansible 2.9.24
  config file = /etc/ansible/ansible.cfg
  configured module search path = [u'/home/admin/.ansible/plugins/modules',
u'/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python2.7/site-packages/ansible
  executable location = /usr/bin/ansible
  python version = 2.7.5 (default, Apr  2 2020, 13:16:51) [GCC 4.8.5 20150623 (Red Hat 4.8.5-39)]
```

（注） 前出の出力が示すように、Ansible が Python 2.x を参照するのは正常です。コネクタは引き続き Python 3 を使用します。

次のタスク

Cisco Secure 動的属性コネクタ をインストールします （11 ページ）で説明されているように、コネクタをインストールします。

オプションで 動的属性コネクタ でのプロキシの使用を停止するには、`/etc/environment` を編集してプロキシ構成を削除します。

前提条件ソフトウェアのインストール：RHEL

始める前に

次のことをすべて行います。

- システムが サポートされているオペレーティングシステムとサードパーティソフトウェア （5 ページ） で説明した前提条件を満たしていることを確認します。
- （オプション）動的属性コネクタ へのプロキシアクセスが必要な場合は、前提条件ソフトウェアのインストール （6 ページ）を参照してください。

ステップ 1 Docker がインストールされていないことを確認し、インストールされている場合はアンインストールします。

```
docker --version
```

Docker がインストールされている場合は、Ubuntu での Docker エンジンのアンインストールの説明に従ってアンインストールします。

ステップ 2 リポジトリを更新します。

RHEL 7：

```
sudo yum -y update && sudo yum -y upgrade
```

RHEL 8：

```
sudo dnf -y update && sudo dnf -y upgrade
```

ステップ 3 epel リポジトリをインストールします。

RHEL 7：

```
sudo yum -y install https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm
```

RHEL 8：

```
sudo dnf -y install https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm
```

ステップ 4 （RHEL 7 のみ。）Python3 をインストールします。

```
sudo yum install -y python3 libselinux-python3
```

ステップ 5 Ansible をインストールします。

RHEL 7：

```
sudo yum -y install ansible
```

RHEL 8：

```
sudo dnf install -y ansible
```

ステップ 6 Ansible のバージョンを確認します。

```
ansible --version
```

次に例を示します。

RHEL 7：

```
ansible 2.9.24
  config file = /etc/ansible/ansible.cfg
  configured module search path = [u'/home/stevej/.ansible/plugins/modules',
u'/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python2.7/site-packages/ansible
  executable location = /usr/bin/ansible
  python version = 2.7.5 (default, Mar 20 2020, 17:08:22) [GCC 4.8.5 20150623 (Red Hat 4.8.5-39)]
```

（注） 前出の出力が示すように、Ansible が Python 2.x を参照するのは正常です。コネクタは引き続き Python 3 を使用します。

RHEL 8：

```
ansible 2.9.24
  config file = /etc/ansible/ansible.cfg
  configured module search path = ['/home/stevej/.ansible/plugins/modules',
'/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python3.6/site-packages/ansible
  executable location = /usr/bin/ansible
  python version = 3.6.8 (default, Mar 18 2021, 08:58:41) [GCC 8.4.1 20200928 (Red Hat 8.4.1-1)]
```

次のタスク

Cisco Secure 動的属性コネクタ をインストールします （11 ページ）で説明されているように、コネクタをインストールします。

オプションで 動的属性コネクタ でのプロキシの使用を停止するには、`/etc/environment` を編集してプロキシ構成を削除します。

前提条件ソフトウェアのインストール：Ubuntu

このタスクでは、Ubuntu に前提条件のソフトウェアをインストールする方法について説明します。

ステップ 1 Docker がインストールされていないことを確認し、インストールされている場合はアンインストールします。

```
docker --version
```

Docker がインストールされている場合は、Ubuntu での Docker エンジンのアンインストールの説明に従ってアンインストールします。

ステップ 2 リポジトリを更新します。

```
sudo apt -y update && sudo apt -y upgrade
```

ステップ 3 Python のバージョンを確認します。

```
/usr/bin/python3 --version
```

バージョンが 3.6 より前の場合は、バージョン 3.6 以降をインストールする必要があります。

ステップ 4 Python 3.6 をインストールします。

```
sudo apt -y install python3.6
```

ステップ 5 共通ライブラリをインストールします。

```
sudo apt -y install software-properties-common
```

ステップ 6 Ansible をインストールします。

```
sudo apt-add-repository -y -u ppa:ansible/ansible && sudo apt -y install ansible
```

ステップ 7 Ansible のバージョンを確認します。

```
ansible --version
```

次に例を示します。

```
ansible --version
ansible 2.9.19
  config file = /etc/ansible/ansible.cfg
  configured module search path = [u'/home/admin/.ansible/plugins/modules',
u'/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python2.7/dist-packages/ansible
  executable location = /usr/bin/ansible
  python version = 2.7.17 (default, Feb 27 2021, 15:10:58) [GCC 7.5.0]
```

（注） 前出の出力が示すように、Ansible が Python 2.x を参照するのは正常です。コネクタは引き続き Python 3.6 を使用します。

次のタスク

Cisco Secure 動的属性コネクタ をインストールします （11 ページ）で説明されているように、コネクタをインストールします。

オプションで 動的属性コネクタ でのプロキシの使用を停止するには、`/etc/environment` を編集してプロキシ構成を削除します。

Cisco Secure 動的属性コネクタ をインストールします

インストールについて

このトピックでは、Cisco Secure 動的属性コネクタ のインストールについて説明します。`sudo` 権限を持つユーザーとしてコネクタをインストールする必要がありますが、非権限ユーザーとしてコネクタを実行できます。

はじめる前に

システムに次の前提条件ソフトウェアがインストールされていることを確認してください。

- Ubuntu 18.04 ～ 22.04.2
- CentOS 7 Linux または 8
- Red Hat Enterprise Linux（RHEL）7 または 8
- Python 3.6.x 以降
- Ansible 2.9 以降

すべてのオペレーティングシステムの最小要件：

- 4 個の CPU
- 8 GB RAM
- 新規インストールの場合は、100GB の空きディスク容量

次のように仮想マシンのサイズを設定することを推奨します。

- 50 個のコネクタ（コネクタごとに 5 つのフィルタと 20,000 ワークロードを想定）、4 つの CPU、8GB RAM、100 GB の空きディスク容量
- 125 個のコネクタ（コネクタごとに 5 つのフィルタと 50,000 ワークロードを想定）、8 つの CPU、16GB RAM、100 GB の空きディスク容量

（注） 仮想マシンのサイズを適切に設定しないと、動的属性コネクタに障害が発生したり、起動しなかったりする可能性があります。

vCenter 属性を使用する場合は、次も必要です。

- vCenter 6.7
- 仮想マシンに、VMware ツールがインストールされている必要があります。

前提条件ソフトウェアをインストールするには、前提条件ソフトウェアのインストール （6 ページ）を参照してください。

Readme とリリース ノートの表示

最新のインストール情報については、以下を参照してください。

Readme：https://galaxy.ansible.com/cisco/csdac

リリースノート：Cisco Secure 動的属性コネクタ リリースノート

動的属性コネクタソフトウェアの取得

動的属性コネクタ ソフトウェアの最新バージョンを取得するには、次のコマンドを実行します。

```
ansible-galaxy collection install cisco.csdac
```

ムスター（収集）サービスのインストール

ムスター（収集）サービスは、動的属性コネクタ の別名です。

~/.ansible/collections/ansible_collections/cisco/csdac ディレクトリから次のコマンドを実行します。

ansible-playbook default_playbook.yml [--ask-become-pass] [--extra-vars " *vars* "]

構文の説明

--ask-become-pass	**sudo** パスワードを入力するように求められます。マシンで sudo が有効になっている場合は必須です。

--extra-vars

次のオプションの追加変数により、動的属性コネクタ がプロキシを使用できるようになります。使用する値は、前提条件ソフトウェアのインストール （6 ページ）の説明に従って構成した /etc/environment の値と一致する必要があります。

- **csdac_proxy_enabled=true**
- **csdac_http_proxy_url=http://***PROXY_URL*

  csdac_https_proxy_url=*PROXY_URL*

次のオプションの追加変数は、動的属性コネクタ に安全に接続するために使用できる自己署名証明書を作成します。これらのパラメータを省略すると、動的属性コネクタ はデフォルトの証明書を使用します。

- **csdac_certificate_domain**

  自動生成された証明書のドメイン名。デフォルト値は、ホストの自動検出されたホスト名です (ansible によって検出される)

- **csdac_certificate_country_name**

  Two-letter country code. （デフォルト値は `us`）

- **csdac_certificate_organization_name**

  組織名。（デフォルト値は `cisco`）

- **csdac_certificate_organization_unit_name**
- 組織単位名（デフォルト値は `Cisco`）)

デフォルトの証明書を使用したインストール例

たとえば、デフォルトのオプションでソフトウェアをインストールするには：

```
ansible-galaxy collection install cisco.csdac
cd ~/.ansible/collections/ansible_collections/cisco/csdac
ansible-playbook default_playbook.yml --ask-become-pass
```

オプションの証明書を使用したインストール例

たとえば、オプションの証明書を使用してソフトウェアをインストールするには：

```
ansible-galaxy collection install cisco.csdac
cd ~/.ansible/collections/ansible_collections/cisco/csdac
ansible-playbook default_playbook.yml --ask-become-pass --extra-vars
"csdac_certificate_domain=domain.example.com csdac_certificate_country_name=US
csdac_certificate_organization_name=Cisco
csdac_certificate_organization_unit_name=Engineering"
```

証明書を作成したら、コネクタへのアクセスに使用する Web ブラウザに証明書をインポートします。証明書は `~/csdac/app/config/certs` ディレクトリに作成されます。

インストールログの表示

インストールログは次の場所にあります。

```
~/.ansible/collections/ansible_collections/cisco/csdac/logs/csdac.log
```

証明書を使用した 動的属性コネクタ への接続

証明書とキーがある場合は、それらを仮想マシンの `~/csdac/app/config/certs` ディレクトリに配置します。

前のタスクを実行した後、次のコマンドを入力して、動的属性コネクタ の Docker コンテナを再起動します。

```
docker restart muster-ui
```

コネクタへのログイン

1. `https://ip-address` で 動的属性コネクタ にアクセスします。
2. ログインします。

   初回ログインのユーザー名は `admin`、パスワードは `admin` です。初めてログインしたときに、パスワードを変更するよう求められます。

Cisco Secure 動的属性コネクタのアップグレード

このトピックでは、以前の Cisco Secure 動的属性コネクタ を現在のバージョンにアップグレードする方法について説明します。これらのタスクは、Cisco Secure 動的属性コネクタ のバージョンやオペレーティングシステムに関係なく実行できます。

ステップ 1 アップグレードするマシンにログインします。

ステップ 2 次のコマンドを入力します。

```
cd ~/.ansible/collections/ansible_collections/cisco/csdac
ansible-galaxy collection install cisco.csdac --force
ansible-playbook default_playbook.yml --ask-become-pass [--extra-vars vars]
```

構文の説明

--ask-become-pass	**sudo** パスワードを入力するように求められます。マシンで sudo が有効になっている場合は必須です。

--extra-vars

次のオプションの追加変数により、動的属性コネクタ がプロキシを使用できるようになります。使用する値は、前提条件ソフトウェアのインストール （6 ページ）の説明に従って構成した /etc/environment の値と一致する必要があります。

- **csdac_proxy_enabled=true**
- **csdac_http_proxy_url=http://***PROXY_URL*

  csdac_https_proxy_url=*PROXY_URL*

次のオプションの追加変数は、動的属性コネクタ に安全に接続するために使用できる自己署名証明書を作成します。これらのパラメータを省略すると、動的属性コネクタはデフォルトの証明書を使用します。

- **csdac_certificate_domain**

  自動生成された証明書のドメイン名。デフォルト値は、ホストの自動検出されたホスト名です (ansible によって検出される)

- **csdac_certificate_country_name**

  Two-letter country code. （デフォルト値は `us`）

- **csdac_certificate_organization_name**

  組織名。（デフォルト値は `cisco`）

- **csdac_certificate_organization_unit_name**
- 組織単位名（デフォルト値は `Cisco`）)

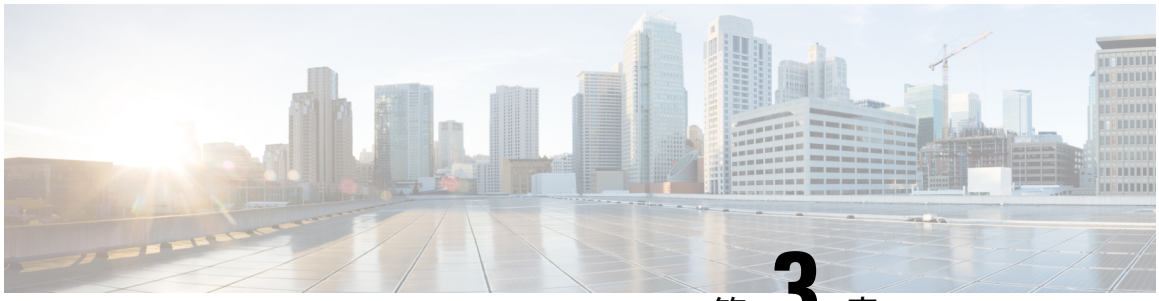
ステップ 3 アップグレードが完了するまで待ちます。

ステップ 4 アップグレードのログは次の場所にあります。

```
~/.ansible/collections/ansible_collections/cisco/csdac/logs/csdac.log
```

次のタスク

コネクタの作成 （17 ページ） を参照してください。



第 3 章

Cisco Secure 動的属性コネクタ の設定

動的属性コネクタをインストールして、コネクタ、動的属性フィルタ、アダプタを構成し、アクセス制御ルールで使用できるダイナミック ネットワーク データを management center に提供します。

動的属性コネクタにより、コネクタを構成して、アクセス制御ルールで使用できるダイナミック ネットワーク データを management center に提供できます。

詳細については、次のトピックを参照してください。

コネクタの作成

コネクタは、クラウドサービスでのインターフェイスです。コネクタはクラウドサービスからネットワーク情報を取得するため、management center のアクセス コントロール ポリシーでネットワーク情報を使用できます。

次がサポートされています。

表 *2 : Cisco Secure* 動的属性コネクタ バージョンおよびプラットフォーム でサポートされているコネクタのリスト

CSDAC バージョン/プラットフォーム	AWS	GitHub	Google クラウド	Azure	Azure サービスタグ	Microsoft Office 365	vCenter
バージョン 1.1（オンプレミス）	対応	×	×	対応	対応	対応	対応
バージョン 2.0（オンプレミス）	対応	対応	対応	対応	対応	対応	対応
バージョン 2.2（オンプレミス）	対応	対応	対応	対応	対応	対応	対応

詳細については、次の項を参照してください。

Amazon Web Services コネクタ：ユーザー権限とインポートされたデータについて

Cisco Secure 動的属性コネクタ は、アクセス コントロール ポリシーで使用するために AWS から management center に動的属性をインポートします。

インポートされた動的属性

AWS から次の動的属性をインポートします。

- タグ：AWS EC2 リソースを整理するために使用できるユーザー定義のキーと値のペア。

  詳細については、AWS ドキュメントの「Amazon EC2 リソースのタグ付け」を参照してください。

- AWS 内の仮想マシンの *IP* アドレス。

必要最小限の権限

Cisco Secure 動的属性コネクタ には、少なくとも、`ec2:DescribeTags` および `ec2:DescribeInstances` に動的属性のインポートを許可するポリシーを持つユーザーが必要です。

Cisco Secure 動的属性コネクタ に対して最小限の権限を持つ AWS ユーザーを作成します。

このタスクでは、動的属性を management center に送信するための最小限の権限を持つサービスアカウントを設定する方法について説明します。これらの属性のリストについては、Amazon Web Services コネクタ：ユーザー権限とインポートされたデータについて （18 ページ）を参照してください。

始める前に

Amazon Web Services（AWS）アカウントがすでに設定されている必要があります。これを行う方法の詳細については、AWS ドキュメントのこの記事を参照してください。

ステップ 1 管理者ロールを持つユーザーとして AWS コンソールにログインします。

ステップ 2 ダッシュボードから、**[セキュリティ、アイデンティティおよび遵守（Security, Identity & Compliance）]** > **[IAM]**をクリックします。

ステップ 3 **[アクセス管理（Access Management）]** > **[ユーザー（Users）]**をクリックします。

ステップ 4 [ユーザの追加（Add Users）] をクリックします。

ステップ 5 [ユーザー名（User Name）] フィールドに、ユーザーを識別するための名前を入力します。

ステップ 6 [アクセスキー - プログラムによるアクセス（Access Key - Programmatic Access）] をクリックします。

ステップ 7 [権限の設定（Set permissions）] ページで、ユーザーに何もアクセスを許可せずに [次へ（Next）] をクリックします。これは後で行います。

ステップ 8 必要に応じて、ユーザーにタグを追加します。

ステップ 9 [Create User] をクリックします。

ステップ 10 [.csvをダウンロード（Download.csv）] をクリックして、ユーザーのキーをコンピューターにダウンロードします。

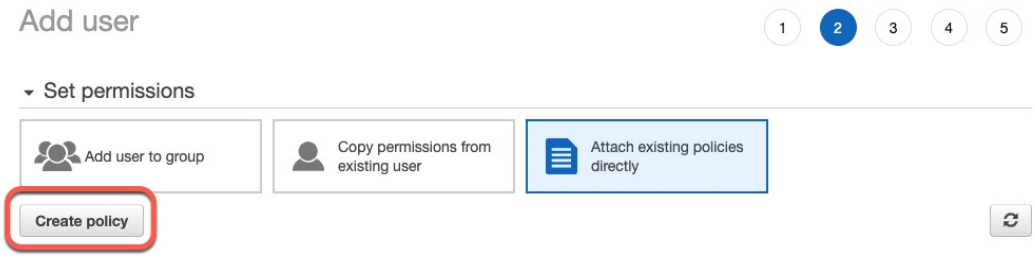
（注） これが、ユーザーのキーを取得する必要がある唯一の機会です。

ステップ 11 [閉じる（Close）] をクリックします。

ステップ 12 左側の列の [アイデンティティとアクセス管理（IAM）（Identity and Access Management (IAM)）] ページで、**[アクセス管理（Access Management）] > [ポリシー（Policies）]**をクリックします。

ステップ 13 [Create Policy] をクリックします。

ステップ 14 [ポリシーの作成（Create Policy）] ページで、[JSON] をクリックします。



ステップ 15 フィールドに次のポリシーを入力します。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeTags",
                "ec2:DescribeInstances"
            ],
            "Resource": "*"
        }
    ]
}
```

ステップ 16 [次へ（Next）] をクリックします。

ステップ 17 [レビュー（Review）] をクリックします。

ステップ 18 [ポリシーの確認（Review Policy）] ページで、必要な情報を入力し、[ポリシーの作成（Create Policy）] をクリックします。

ステップ 19 [ポリシー（Policies）]ページで、検索フィールドにポリシー名のすべてまたは一部を入力し、Enter キーを押します。

ステップ 20 作成したポリシーをクリックします。

ステップ 21 **[アクション（Actions）] > [アタッチ（Attach）]**をクリックします。

ステップ 22 必要に応じて、検索フィールドにユーザー名の全部または一部を入力し、Enter キーを押します。

ステップ 23 [ポリシーをアタッチ（Attach policy）] をクリックします。

次のタスク

AWS コネクタの作成 （20 ページ）。

AWS コネクタの作成

このタスクでは、アクセス コントロール ポリシーで使用するため、AWS から management center にデータを送信するコネクタを設定する方法について説明します。

始める前に

Cisco Secure 動的属性コネクタ に対して最小限の権限を持つ AWS ユーザーを作成します。（18 ページ） で説明した権限以上のユーザーを作成します。

ステップ 1 動的属性コネクタにログインします。

ステップ 2 [コネクタ（Connectors）] をクリックします。

ステップ 3 次のいずれかを実行します。

- 新しいコネクタの追加：[追加（Add）]アイコン（  ）をクリックしてから、コネクタの名前をクリックします。
- コネクタの編集または削除：**その他**（  ）をクリックしてから、行の末尾にある [編集（Edit）] または [削除（Delete）] をクリックします。

ステップ 4 次の情報を入力します。

値	説明
名前（Name）	（必須）このコネクタを一意に識別する名前を入力します。
説明（Description）	説明（オプション）。
プル間隔（Pull Interval）	（デフォルトは 30 秒）AWS から IP マッピングを取得する間隔です。
リージョン（Region）	（必須）AWS リージョンコードを入力します。
アクセスキー（Access Key）	（必須）アクセスキーを入力します。
秘密キー（Secret Key）	（必須）秘密鍵を入力します。

ステップ 5 コネクタを保存する前に、[テスト（Test）] をクリックして、テストが成功することを確認します。

ステップ 6 [保存（Save）] をクリックします。

ステップ 7 [ステータス（Status）] 列に [OK] が表示されていることを確認します。

Azure コネクタ：ユーザー権限とインポートされたデータについて

Cisco Secure 動的属性コネクタ は、アクセス コントロール ポリシーで使用するために、Azure から management center へ動的属性をインポートします。

インポートされた動的属性

Azure から次の動的属性をインポートします。

- タグ：リソース、リソースグループ、およびサブスクリプションに関連付けられたキーと値のペア。

  詳細については、Microsoft ドキュメントのこのページを参照してください。

- Azure 内の仮想マシンの *IP* アドレス。

必要な最小限の権限

Cisco Secure 動的属性コネクタ で、動的属性をインポートするには、少なくとも**リーダー**権限を持つユーザーが必要です。

Cisco Secure 動的属性コネクタ に対する最小限の権限を持つ Azure ユーザーの作成

このタスクでは、動的属性を management center に送信するための最小限の権限を持つサービスアカウントを設定する方法について説明します。これらの属性のリストについては、Azure コネクタ：ユーザー権限とインポートされたデータについて （21 ページ）を参照してください。

始める前に

Microsoft Azure アカウントを既に持っている必要があります。設定するには、Azure ドキュメントサイトのこのページを参照してください。

ステップ 1 サブスクリプションの所有者として Azure Portal にログインします。

ステップ 2 **[Azure Active Directory]** をクリックします。

ステップ 3 設定するアプリケーションの Azure Active Directory のインスタンスを見つけます。

ステップ 4 **[追加（Add）]** > **[アプリケーションの登録（App registration）]** をクリックします。

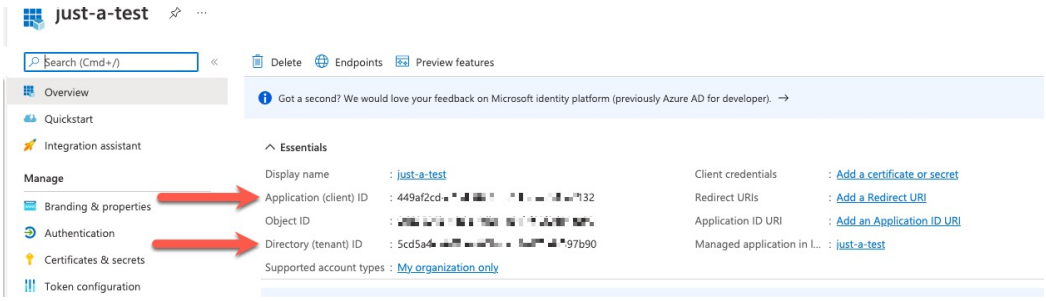
ステップ 5 [名前（Name）] フィールドに、このアプリケーションを識別するための名前を入力します。

ステップ 6 組織の必要に応じて、このページにその他の情報を入力します。

ステップ 7 [登録（Register）] をクリックします。

ステップ 8 次のページで、クライアント ID (アプリケーション *ID* とも呼ばれる) とテナント ID (ディレクトリ *ID* とも呼ばれる) を書き留めます。

次に例を示します。

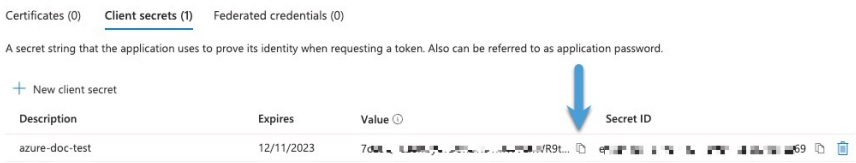


ステップ 9 [クライアントクレデンシャル（Client Credentials）] の横にある [証明書またはシークレットの追加（Add a certificate or secret）] をクリックします。

ステップ 10 [新しいクライアントシークレット（New Client Secret）] をクリックします。

ステップ 11 要求された情報を入力し、[追加（Add）] をクリックします。

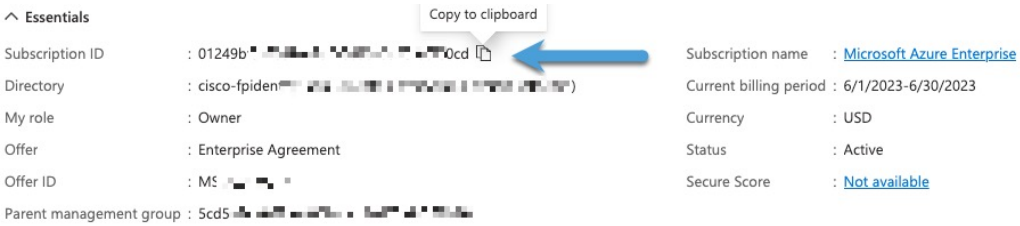
ステップ 12 [値（Value）] フィールドの値をクリップボードにコピーします。[シークレットID（Secret ID）] ではなく、この値がクライアントシークレットです。



ステップ 13 Azure Portal のメインページに戻り、[サブスクリプション（Subscriptions）] をクリックします。

ステップ 14 サブスクリプションの名前をクリックします。

ステップ 15 クリップボードにサブスクリプション ID をコピーします。



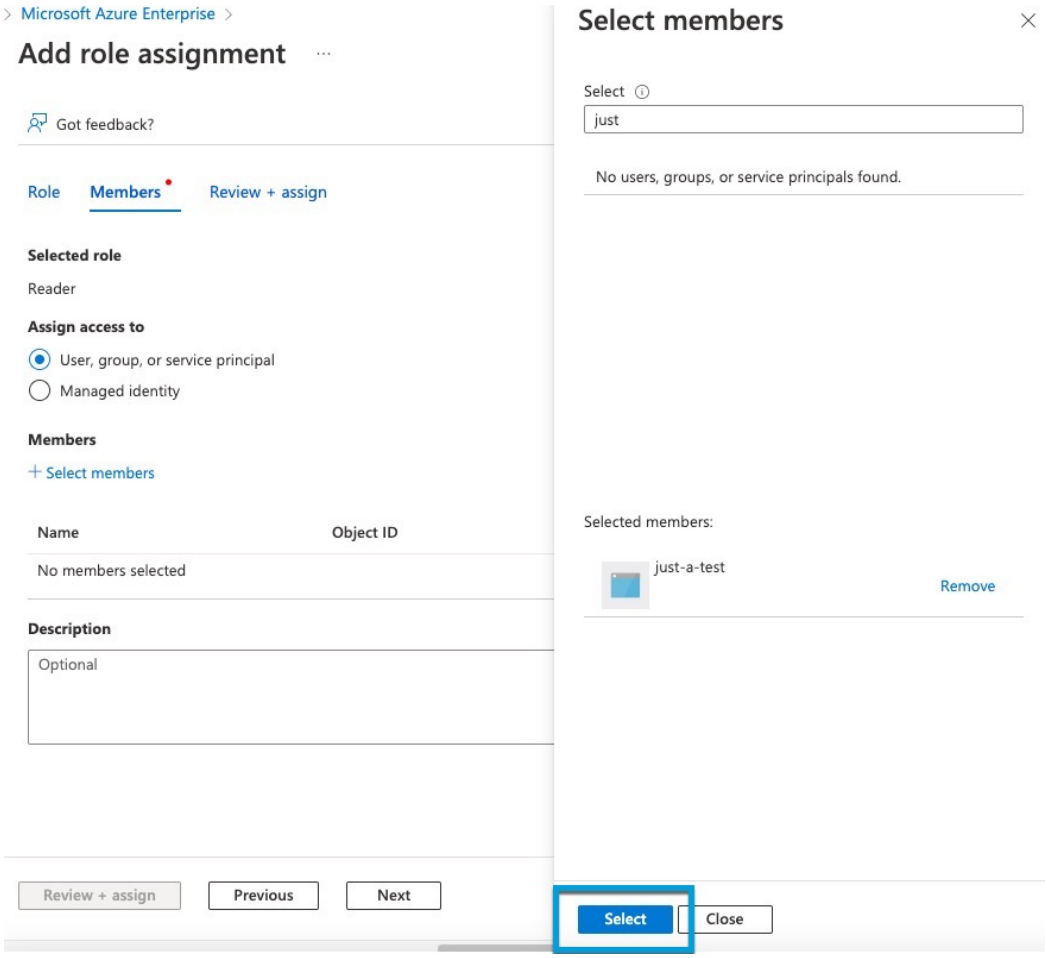
ステップ 16 [アクセス制御（IAM）（Access Control (IAM)）] をクリックします。

ステップ 17 **[追加（Add）] > [ロール割り当ての追加（Add role assignment）]**をクリックします。

ステップ 18 [リーダー（Reader）] をクリックし、[次へ（Next）] をクリックします。

ステップ 19 [メンバーの選択（Select Members）]をクリックします。

ステップ 20 ページの右側で、登録したアプリケーションの名前をクリックし、[選択（Select）] をクリックします。



ステップ 21 [確認と割り当て（Review + Assign）] をクリックし、プロンプトに従って操作を完了します。

次のタスク

Azure コネクタの作成 （23 ページ） を参照してください。

Azure コネクタの作成

このタスクでは、アクセス コントロール ポリシーで使用するために Azure から management center にデータを送信するコネクタを作成する方法について説明します。

始める前に

Cisco Secure 動的属性コネクタ に対する最小限の権限を持つ Azure ユーザーの作成 （21 ページ） で説明した権限以上の Azure ユーザーを作成します。

ステップ 1 動的属性コネクタにログインします。

ステップ 2 [コネクタ（Connectors）] をクリックします。

ステップ 3 次のいずれかを実行します。

- 新しいコネクタの追加：[追加（Add）]アイコン（  ）をクリックしてから、コネクタの名前をクリックします。
- コネクタの編集または削除：**その他**（  ）をクリックしてから、行の末尾にある [編集（Edit）] または [削除（Delete）] をクリックします。

ステップ 4 次の情報を入力します。

値	説明
名前（Name）	（必須）このコネクタを一意に識別する名前を入力します。
説明（Description）	説明（オプション）。
プル間隔（Pull Interval）	（デフォルトは 30 秒）Azure から IP マッピングを取得する間隔です。
サブスクリプションID（Subscription Id）	（必須）Azure サブスクリプション ID を入力します。
テナントID（Tenant Id）	（必須）テナント ID を入力します。
クライアント ID（Client Id）	（必須）クライアント ID を入力します。
クライアントのシークレット（Client Secret）	（必須）クライアントのシークレットを入力します。

ステップ 5 コネクタを保存する前に、[テスト（Test）] をクリックして、**Test connection succeeded** が表示されることを確認します。

ステップ 6 [保存（Save）] をクリックします。

ステップ 7 [ステータス（Status）] 列に [OK] が表示されていることを確認します。

Azure サービスタグコネクタの作成

このトピックでは、アクセス コントロール ポリシーで使用する management center への Azure サービスタグのコネクタを作成する方法について説明します。これらのタグに関連付けられた IP アドレスは、Microsoft によって毎週更新されます。

詳細については、Microsoft TechNet の「仮想ネットワーク サービス タグ」を参照してください。

ステップ 1 動的属性コネクタにログインします。

ステップ 2 [コネクタ（Connectors）] をクリックします。

ステップ 3 次のいずれかを実行します。

- 新しいコネクタの追加：[追加（Add）]アイコン（）をクリックしてから、コネクタの名前をクリックします。
- コネクタの編集または削除：**その他**（  ）をクリックしてから、行の末尾にある [編集（Edit）] または [削除（Delete）] をクリックします。

ステップ 4 次の情報を入力します。

値	説明
名前（Name）	（必須）このコネクタを一意に識別する名前を入力します。
説明（Description）	説明（オプション）。
プル間隔（Pull Interval）	（デフォルトは 30 秒）Azure から IP マッピングを取得する間隔です。
サブスクリプションID（Subscription Id）	（必須）Azure サブスクリプション ID を入力します。
テナントID（Tenant Id）	（必須）テナント ID を入力します。
クライアント ID（Client Id）	（必須）クライアント ID を入力します。
クライアントのシークレット（Client Secret）	（必須）クライアントのシークレットを入力します。

ステップ 5 コネクタを保存する前に、[テスト（Test）] をクリックして、**Test connection succeeded** が表示されることを確認します。

ステップ 6 [保存（Save）] をクリックします。

ステップ 7 [ステータス（Status）] 列に [OK] が表示されていることを確認します。

GitHub コネクタの作成

このセクションでは、アクセス コントロール ポリシーで使用するためにデータを management center に送信する GitHub コネクタを作成する方法について説明します。これらのタグに関連付けられている IP アドレスは、GitHub によって管理されています。動的属性フィルタを作成する必要はありません。

詳細については、「GitHub の IP アドレスについて」を参照してください。

（注） IP アドレスの取得に失敗するため、URL は変更しないでください。

ステップ 1 動的属性コネクタ にログインします。

ステップ 2 [コネクタ（Connectors）] をクリックします。

ステップ 3 次のいずれかを実行します。

- 新しいコネクタの追加：[追加（Add）]アイコン（  ）をクリックしてから、コネクタの名前をクリックします。
- コネクタの編集または削除：**その他**（  ）をクリックしてから、行の末尾にある [編集（Edit）] または [削除（Delete）] をクリックします。

ステップ 4 [名前（Name）] とオプションの [説明（Description）]を入力します。

ステップ 5 （オプション）[プル間隔（Pull Interval）] フィールドで、動的属性コネクタが GitHub から IP アドレスを取得する頻度を秒単位で変更します。デフォルトは 21,600 秒（6 時間）です。

ステップ 6 コネクタを保存する前に、[テスト（Test）] をクリックして、テストが成功することを確認します。

ステップ 7 [保存（Save）] をクリックします。

ステップ 8 [ステータス（Status）] 列に [OK] が表示されていることを確認します。

Google Cloud コネクタ：ユーザー権限とインポートされたデータについて

Cisco Secure 動的属性コネクタ は、アクセス コントロール ポリシーで使用するために、Google Cloud から management center へ動的属性をインポートします。

インポートされた動的属性

次の動的属性を Google Cloud からインポートします。

- ラベル：Google Cloud リソースを整理するために使用できるキーと値のペア。

  詳細については、Google Cloud ドキュメントの「ラベルの作成と管理」を参照してください。

- ネットワークタグ：組織、フォルダー、またはプロジェクトに関連付けられたキーと値のペア。

  詳細については、Google Cloud ドキュメントの「タグの作成と管理」を参照してください。

- Google Cloud 内の仮想マシンの *IP* アドレス。

必要最小限の権限

Cisco Secure 動的属性コネクタ では、少なくとも、動的属性をインポートできる基本閲覧者（Basic Viewer）権限を持つユーザーが必要です。 >

Cisco Secure 動的属性コネクタ に対して最小限の権限を持つ Google Cloud ユーザーを作成します。

このタスクでは、動的属性を management center に送信するための最小限の権限を持つサービスアカウントを設定する方法について説明します。これらの属性のリストについては、Google Cloud コネクタ：ユーザー権限とインポートされたデータについて （26 ページ）を参照してください。

始める前に

Google Cloud アカウントがすでに設定されている必要があります。設定方法に関する詳細情報については、 Google Cloud ドキュメントの「環境設定」を参照してください。

ステップ 1 所有者ロールを持つユーザーとして Google Cloud アカウントにログインします。

ステップ 2 **[IAMおよび管理者（IAM & Admin）] > [サービスアカウント（Service Accounts）] > [サービスアカウントの作成（Create Service Account）]** をクリックします。

ステップ 3 次の情報を入力します。

- サービスアカウント名（Service account name） ：このアカウントを識別するための名前。たとえば、**CSDAC**。
- サービスアカウントID（Service account ID） ：サービスアカウント名を入力した後、一意の値を入力する必要があります。
- サービスアカウントの説明（Service account description） ： オプションの説明を入力します。

サービスアカウントの詳細については、Google Cloud ドキュメントの「サービス アカウントについて」を参照してください。

ステップ 4 [作成して続行（Create and Continue）] をクリックします。

ステップ 5 [このサービスアカウントへのアクセスをユーザーに許可する（Grant users access to this service account）] セクションが表示されるまで、画面の指示に従います。

ステップ 6 ユーザーに基本閲覧者（Basic Viewer）ロールを付与します。 >

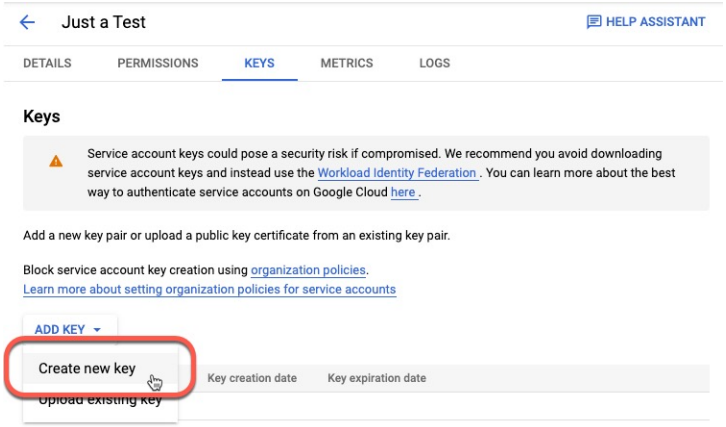
ステップ 7 [完了（Done）] をクリックします。

サービスアカウントのリストが表示されます。

ステップ 8 作成したサービスアカウントの行の末尾にある **その他** （⋮） をクリックします。

ステップ 9 [キーの管理（Manage Keys）] をクリックします。

ステップ 10 **[キーの追加（ADD KEY）] > [新しいキーの作成（Create New Key）]**をクリックします。



ステップ 11 [JSON] をクリックします。

ステップ 12 [作成（Create）] をクリックします。

JSON キーがコンピュータにダウンロードされます。

ステップ 13 GCP コネクタを構成するときは、キーを手元に置いておいてください。

次のタスク

Google Cloud コネクタの作成 （28 ページ） を参照してください。

Google Cloud コネクタの作成

始める前に

Google Cloud JSON 形式のサービスアカウントデータを準備します。コネクタの設定に必要です。

ステップ 1 動的属性コネクタにログインします。

ステップ 2 [コネクタ（Connectors）] をクリックします。

ステップ 3 次のいずれかを実行します。

- 新しいコネクタの追加：[追加（Add）]アイコン（＋） をクリックしてから、コネクタの名前をクリックします。
- コネクタの編集または削除：**その他**（⋮） をクリックしてから、行の末尾にある [編集（Edit）] または [削除（Delete）] をクリックします。

ステップ 4 次の情報を入力します。

値	説明
名前（Name）	（必須）このコネクタを一意に識別する名前を入力します。

値	説明
説明（Description）	説明（オプション）。
プル間隔（Pull Interval）	（デフォルトは 30 秒）AWS から IP マッピングを取得する間隔です。
GCP リージョン（GCP region）	（必須）Google Cloud が配置されている GCP リージョンを入力します。詳細については、Google Cloud のドキュメント「リージョンとゾーン」を参照してください。
サービス アカウント	Google Cloud サービスアカウントの JSON コードを貼り付けます。

ステップ 5 コネクタを保存する前に、[テスト（Test）] をクリックして、テストが成功することを確認します。

ステップ 6 [保存（Save）] をクリックします。

ステップ 7 [ステータス（Status）] 列に [OK] が表示されていることを確認します。

Office 365 コネクタの作成

このタスクでは、アクセス コントロール ポリシーで使用するためのデータを management center に送信する、Office 365 タグのコネクタを作成する方法について説明します。これらのタグに関連付けられた IP アドレスは、Microsoft によって毎週更新されます。データを使用するために動的属性フィルタを作成する必要はありません。

詳細については、docs.microsoft.com の「Office 365 URL および IP アドレス範囲」を参照してください。

ステップ 1 動的属性コネクタにログインします。

ステップ 2 [コネクタ（Connectors）] をクリックします。

ステップ 3 次のいずれかを実行します。

- 新しいコネクタの追加：[追加（Add）]アイコン（＋）をクリックしてから、コネクタの名前をクリックします。
- コネクタの編集または削除：**その他**（⋮）をクリックしてから、行の末尾にある [編集（Edit）] または [削除（Delete）] をクリックします。

ステップ 4 次の情報を入力します。

値	説明
名前（Name）	（必須）このコネクタを一意に識別する名前を入力します。
説明（Description）	説明（オプション）。
プル間隔（Pull Interval）	（デフォルトは 30 秒）Azure から IP マッピングを取得する間隔です。

値	説明
ベース API URL（Base API URL）	（必須）デフォルトと異なる場合は、Office 365 情報を取得する URL を入力します。詳細については、Microsoft ドキュメントサイトの「Office 365 IP アドレスと URL の Web サービス」を参照してください。
インスタンス名（Instance name）	（必須）リストからインスタンス名をクリックします。詳細については、Microsoft ドキュメントサイトの「Office 365 IP アドレスと URL の Web サービス」を参照してください。
オプションの IP を無効にする	（必須）**true**または**false**の入力。

ステップ 5 コネクタを保存する前に、[テスト（Test）] をクリックして、テストが成功することを確認します。

ステップ 6 [保存（Save）] をクリックします。

ステップ 7 [ステータス（Status）] 列に [OK] が表示されていることを確認します。

vCenter コネクタ：ユーザー権限とインポートされたデータについて

Cisco Secure 動的属性コネクタ は、アクセス コントロール ポリシーで使用するために、vCenter から management center へ動的属性をインポートします。

インポートされた動的属性

vCenter から次の動的属性をインポートします。

- オペレーティング システム
- *MAC* アドレス
- *IP* アドレス
- *NSX* タグ

必要最小限の権限

Cisco Secure 動的属性コネクタ では、少なくとも、動的属性をインポートできる**読み取り専用**権限を持つユーザーが必要です。

vCenter コネクタの作成

このタスクでは、アクセス コントロール ポリシーで使用するためにデータを management center に送信する VMware vCenter のコネクタを作成する方法について説明します。

始める前に

信頼されていない証明書を使用して vCenter と通信する場合は、認証局 (CA) チェーンの手動での取得 （39 ページ）を参照してください。

ステップ 1 動的属性コネクタにログインします。

ステップ 2 [コネクタ（Connectors）] をクリックします。

ステップ 3 次のいずれかを実行します。

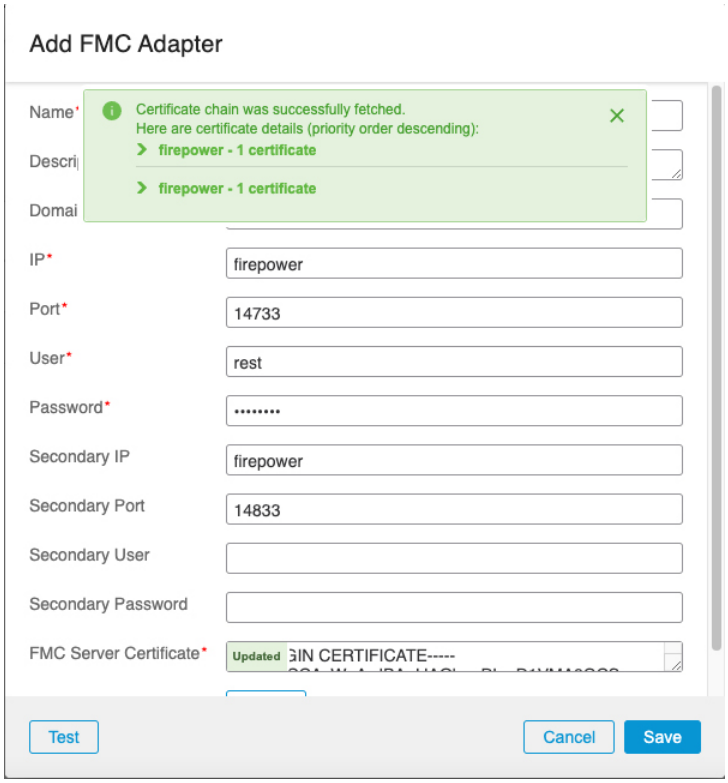
- 新しいコネクタの追加：[追加（Add）]アイコン（  ）をクリックしてから、コネクタの名前をクリックします。
- コネクタの編集または削除：**その他**（  ） をクリックしてから、行の末尾にある [編集（Edit）] または [削除（Delete）] をクリックします。

ステップ 4 次の情報を入力します。

値	説明
名前（Name）	（必須）このコネクタを一意に識別する名前を入力します。
説明（Description）	任意で説明を入力します。
プル間隔（Pull Interval）	（デフォルトは 30 秒）vCenter から IP マッピングを取得する間隔です。
ホスト（Host）	（必須）次のいずれかを入力します。 • vCenter の完全修飾ホスト名 • vCenter の IP アドレス • （オプション）ポート スキーム（**`https://`** など）または末尾のスラッシュを入力しないでください。 たとえば、**`myvcenter.example.com`** または **`192.0.2.100:9090`**
ユーザー（User）	（必須）最低限でも読み取り専用ロールを持つユーザーのユーザー名を入力します。ユーザ名は大文字/小文字を区別します。
パスワード（Password）	（必須）ユーザーのパスワードを入力します。
NSX IP	vCenter Network Security Visualization（NSX）を使用する場合は、その IP アドレスを入力します。
NSXユーザー（NSX User）	最低限でも監査人ロールを持つ NSX ユーザーのユーザー名を入力します。
NSXタイプ（NSX Type）	**NSX-T** を入力します。

値	説明
NSXパスワード（NSX Password）	NSX ユーザーのパスワードを入力します。
vCenter証明書（vCenter Certificate）	次の選択肢があります。 • **[証明書を取得（Get Certificate）]** > **[取得（Fetch）]**をクリックして証明書を自動的に取得するか、それが不可能な場合は、認証局(CA)チェーンの手動での取得 （39 ページ） で説明されているように手動で証明書を取得します。 • **[証明書を取得（Get Certificate）]** > **[ファイルから参照（Browse from file）]**をクリックして、以前にダウンロードした証明書チェーンをアップロードします。

次に、証明書チェーンを正常に取得する例を示します。



ダイアログボックスの上部にある証明書 CA チェーンを展開すると、次のような証明書が表示されます。

この方法で証明書を取得できない場合は、認証局 (CA) チェーンの手動での取得 （39 ページ）で説明されているように、証明書チェーンを手動で取得できます。

ステップ 5 コネクタを保存する前に、[テスト（Test）] をクリックして、**Test connection succeeded** が表示されることを確認します。

ステップ 6 [保存（Save）] をクリックします。

次のタスク

アダプタの作成 （33 ページ）

アダプタの作成

アダプタは、アクセスコントロールポリシーで使用するためにクラウドオブジェクトからネットワーク情報をプッシュする Management Center への安全な接続です。

まず、オプションで認証局チェーンを取得できます。これは、management center に安全に接続するために必要です。

認証局チェーンの取得に必要なものは、management center ホスト名のみです。アダプタを作成するには、ユーザー名、パスワード、およびその他の情報が必要です。

動的属性コネクタの Secure Firewall Management Center ユーザーの作成

動的属性コネクタ アダプタ用に専用の management center ユーザーを作成することを推奨します。専用 management center ユーザーを作成すると、management center からの予期しないログアウトなどの問題を回避できます。これは、動的属性コネクタが REST API を使用して定期的にログインし、新規および更新されたダイナミックオブジェクトで management center を更新するためです。

management center ユーザーには、最低限でもアクセス管理者（Access Admin）権限が必要です。

ステップ 1 まだ management center にログインしていない場合は、ログインします。

ステップ 2 **システム**（⚙） > **[ユーザー（Users）]** をクリックします。

ステップ 3 [ユーザの作成（Create User）] をクリックします。

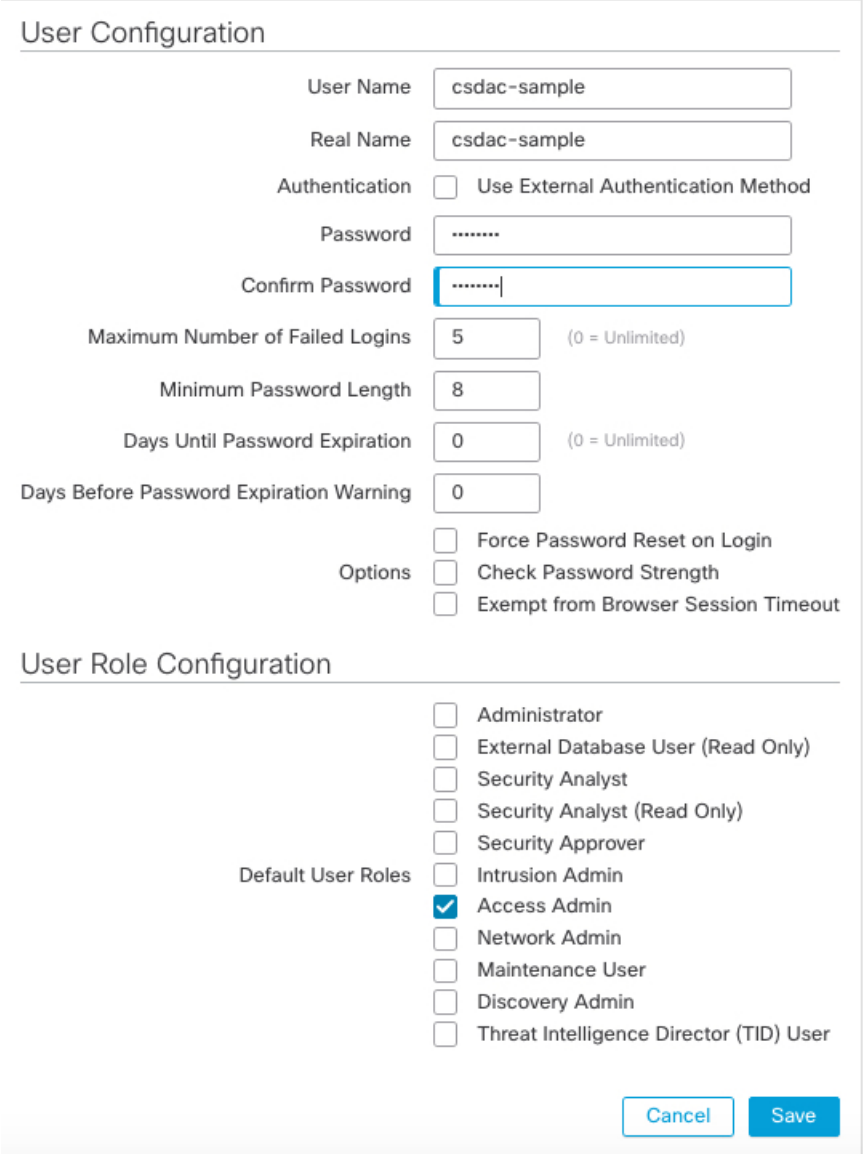
ステップ 4 ユーザーを作成するために必要な情報を入力します。

ステップ 5 [ユーザーロールの構成（User Role Configuration）] で、次のデフォルトロールのいずれか、または同じ権限レベルのカスタムロールをチェックします。

- **管理者（Administrator）**

- アクセス管理者
- ネットワーク管理者

次の図は例を示しています。



REST アクションを許可するために十分な権限を持つカスタムロール、または十分な権限を持つ別のデフォルトロールを選択することもできます。デフォルトロールの詳細については、ユーザーアカウントに関する章の「ユーザーロール」セクションを参照してください。

次のタスク

アダプタの作成 （33 ページ）

オンプレミス Firewall Management Center アダプタを作成する方法

このトピックでは、ダイナミックオブジェクトを 動的属性コネクタ から management center にプッシュするアダプタを作成する方法について説明します。

始める前に

動的属性コネクタの Secure Firewall Management Center ユーザーの作成 （33 ページ）を参照してください。

ステップ 1 動的属性コネクタにログインします。

ステップ 2 [アダプタ（Adapters）] をクリックします。

ステップ 3 次のいずれかを実行します。

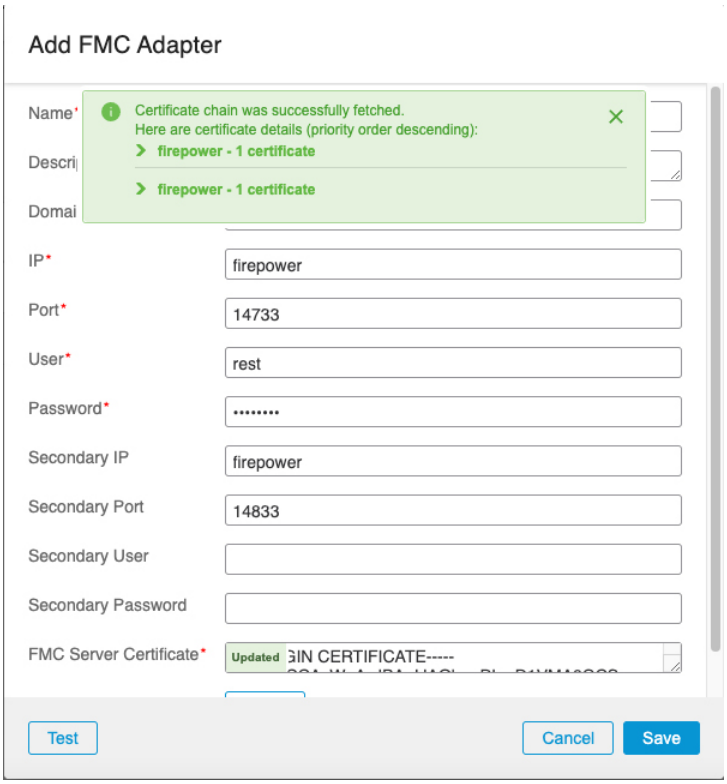
- 新しいコネクタの追加：[追加（Add）]アイコン（  ）をクリックしてから、コネクタの名前をクリックします。
- コネクタの編集または削除：**その他**（  ）をクリックしてから、行の末尾にある [編集（Edit）] または [削除（Delete）] をクリックします。

ステップ 4 次の情報を入力します。

値	説明
名前（Name）	（必須）このアダプタを識別するための一意の名前を入力します。
説明	オプションのアダプタの説明。
ドメイン（Domain）	ダイナミックオブジェクトを作成する Secure Firewall Management Center Virtual ドメインを入力します。グローバルドメインにダイナミックオブジェクトを作成するには、フィールドを空白のままにします。 例：**`Global/MySubdomain`**
IP	（必須）Secure Firewall Management Center Virtual のホスト名または IP アドレスを入力します。 入力するホスト名または IP は、安全に接続するために使用される CA 証明書の共通名と完全に一致している必要があります。
ポート（Port）	（必須）Secure Firewall Management Center Virtual が使用する TLS ポートを入力します。
ユーザー（User）	（必須）最低限でもネットワーク管理者ロールを持つ Secure Firewall Management Center Virtual ユーザーの名前を入力します。
パスワード（Password）	（必須）ユーザーのパスワードを入力します。

値	説明
セカンダリ IP（Secondary IP）	（高可用性のみ。）セカンダリ Secure Firewall Management Center Virtual のホスト名または IP アドレスを入力します。 入力するホスト名または IP は、安全に接続するために使用される CA 証明書の共通名と完全に一致している必要があります。
セカンダリポート（Secondary Port）	（高可用性のみ。）セカンダリ Secure Firewall Management Center Virtual が使用する TLS ポートを入力します。
セカンダリユーザー（Secondary User）	（高可用性のみ。）最低限でもネットワーク管理者ロールを持つセカンダリ Secure Firewall Management Center Virtual ユーザーの名前を入力します。
セカンダリ パスワード（Secondary Password）	（高可用性のみ。）ユーザーのパスワードを入力します。
サーバー証明書	次の選択肢があります。 • **[証明書を取得（Get Certificate）] > [取得（Fetch）]**をクリックして証明書を自動的に取得するか、それが不可能な場合は、認証局(CA)チェーンの手動での取得 （39 ページ） で説明されているように手動で証明書を取得します。 • **[証明書を取得（Get Certificate）] > [ファイルから参照（Browse from file）]**をクリックして、以前にダウンロードした証明書チェーンをアップロードします。

次に、証明書チェーンを正常に取得する例を示します。



ダイアログボックスの上部にある証明書 CA チェーンを展開すると、次のような証明書が表示されます。

この方法で証明書を取得できない場合は、認証局 (CA) チェーンの手動での取得 （39 ページ）で説明されているように、証明書チェーンを手動で取得できます。

ステップ 5 アダプタを保存する前に、[テスト（Test）] をクリックして、テストが成功することを確認します。

ステップ 6 [保存（Save）] をクリックします。

クラウド提供型 Firewall Management Center アダプタの作成

このトピックでは、Cisco Defense Orchestrator で管理される管理センターに、 動的属性コネクタ からダイナミックオブジェクトをプッシュするアダプタを作成する方法について説明します。

クラウド提供型 Firewall Management Center を作成する前に、次のベース URL と API トークンの取得 （38 ページ） の情報を取得してください。

ベース URL と API トークンの取得

このタスクでは、クラウド提供型 Firewall Management Center アダプタの作成に必要な URL と API トークンを CDO から取得する方法について説明します。

始める前に

このセクションで説明するタスクを完了するには、CDO のスーパー管理者である必要があります。

ステップ 1 ネットワーク管理者ロールを持つユーザーとして CDO にログインします。

ステップ 2 ページの右上隅で、[設定（Settings）] をクリックします。

ステップ 3 [全般設定（General Settings）] をクリックします。

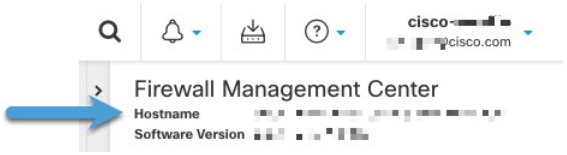
ステップ 4 [APIトークン（API Token）] の横にある [更新（Refresh）] をクリックします。

ステップ 5 後で使用するために、API トークンをテキストファイルにコピーします。

ステップ 6 **[ツールとサービス（Tools & Services）] > [Firewall Management Center]** をクリックします。

ステップ 7 動的属性コネクタ データを送信する管理センターの名前をクリックします。

ステップ 8 [https://] で始まる [ホスト名（Hostname）] の値は、ベース URL です。
次に例を示します。



次のタスク

クラウド提供型 Firewall Management Center アダプタを作成する方法 （38 ページ）。

クラウド提供型 Firewall Management Center アダプタを作成する方法

このタスクでは、CDO によって管理されているデバイスに 動的属性コネクタ からデータを送信する クラウド提供型 Firewall Management Center アダプタを作成する方法について説明します。

始める前に

このタスクを完了する前に、管理センターのベース URL と API トークンを CDO から取得する必要があります。詳細については、ベース URL と API トークンの取得 （38 ページ）を参照してください。

ステップ 1 動的属性コネクタにログインします。

ステップ 2 [アダプタ（Adapters）] をクリックします。

ステップ 3 次のいずれかを実行します。

- 新しいコネクタの追加：[追加（Add）]アイコン（）をクリックしてから、コネクタの名前をクリックします。
- コネクタの編集または削除：**その他**（  ）をクリックしてから、行の末尾にある [編集（Edit）] または [削除（Delete）] をクリックします。

ステップ 4 次の情報を入力します。

値	説明
名前（Name）	（必須）このアダプタを識別するための一意の名前を入力します。
説明	オプションのアダプタの説明。
[ベースURL（Base URL）]	（必須）ベース URL と API トークンの取得 （38 ページ） で見つけたベース URL を使用します。
[API Token]	（必須）ベース URL と API トークンの取得 （38 ページ） で見つけた API トークンを使用します。

ステップ 5 アダプタを保存する前に、[テスト（Test）] をクリックして、テストが成功することを確認します。

ステップ 6 [保存（Save）] をクリックします。

次のタスク

動的属性フィルタの作成 （42 ページ）。

認証局 (CA) チェーンの手動での取得

認証局チェーンを自動的に取得できない場合は、次のブラウザ固有の手順のいずれかを使用して、vCenter、NSX、または Management Center に安全に接続するために使用される証明書チェーンを取得します。

証明書チェーンは、ルート証明書とすべての下位証明書です。

次に接続するには、これらの手順のいずれかを使用する必要があります。

- vCenter または NSX

  Azure または AWS に接続するために証明書チェーンを取得する必要はありません。

- Management Center

この手順を使用する前に、次の認証局チェーンの自動取得に関するセクションを参照してください。

- vCenter コネクタの作成 （30 ページ）

証明書チェーンの取得：Mac（Chrome および Firefox）

Mac OS で Chrome および Firefox ブラウザを使用して証明書チェーンを取得するには、この手順を使用します。

1. ターミナル ウィンドウを開きます。
2. 次のコマンドを入力します。

   ```
   security verify-cert -P url[:port]
   ```

   ここで、url は vCenter または Management Center への URL（スキームを含む）です。次に例を示します。

   ```
   security verify-cert -P https://myvcenter.example.com
   ```

   NAT または PAT を使用して vCenter または Management Center にアクセスする場合は、次のようにポートを追加できます。

   ```
   security verify-cert -P https://myvcenter.example.com:12345
   ```

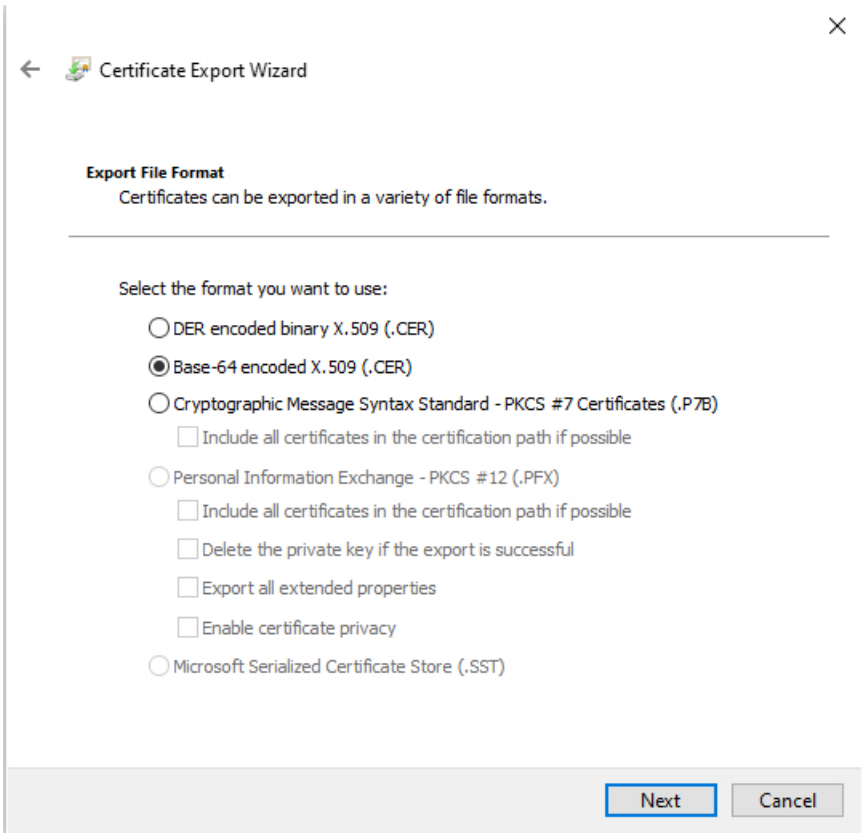
3. 証明書チェーン全体をプレーンテキストファイルに保存します。
   - すべての `-----BEGIN CERTIFICATE-----` および `-----END CERTIFICATE-----` 区切り文字を含めます。
   - 無関係なテキスト（たとえば、証明書の名前、山かっこ（<および>）に含まれるテキスト、および山かっこ自体を除外します。
4. vCenter と Management Center の両方で、これらのタスクを繰り返します。

証明書チェーンの取得：Windows Chrome

Windows で Chrome ブラウザを使用して証明書チェーンを取得するには、この手順を使用します。

1. vCenter または Chrome を使用してログインします。Management Center
2. ブラウザのアドレスバーで、ホスト名の左側にあるロックをクリックします。
3. [証明書（Certificate）] をクリックします。
4. [証明のパス（Certification Path）] タブをクリックします。
5. チェーンの最上位（つまり、最初）の証明書をクリックします。
6. **証明書の表示**をクリックします。
7. [詳細（Details）] タブをクリックします。
8. [ファイルにコピーする（Copy to File）] をクリックします。
9. プロンプトに従って、証明書チェーン全体を含むCER形式の証明書ファイルを作成します。

   エクスポートファイル形式の選択を求められたら、次の図に示すように、[Base 64 エンコード X.509（.CER）（Base 64-Encoded X.509 (.CER)）] をクリックします。



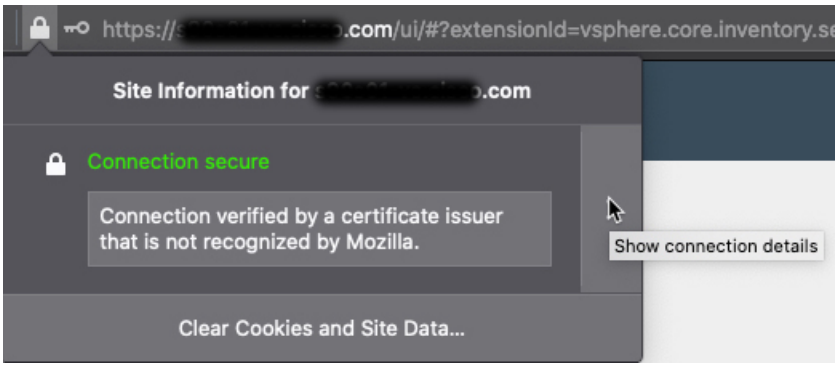
10. 指示に従ってエクスポートを完了します。
11. 証明書をテキストエディタで開きます。
12. チェーン内のすべての証明書に対してこのプロセスを繰り返します。

    テキストエディタに各証明書を最初から最後まで順番に貼り付ける必要があります。
13. vCenter と FMC の両方でこれらのタスクを繰り返します。

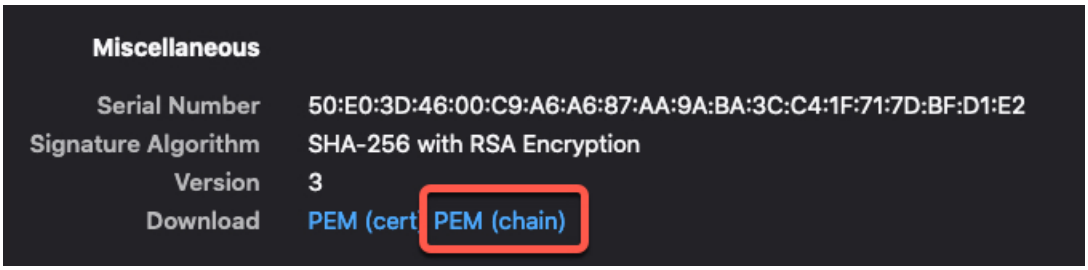
証明書チェーンの取得：Windows Firefox

Windows または Mac OS で Firefox ブラウザの証明書チェーンを取得するには、次の手順を使用します。

1. Firefox を使用して vCenter または Management Center にログインします。
2. ホスト名の左側にあるロックをクリックします。
3. 右矢印（[接続の詳細を表示（Show connection details）]）をクリックします。次の図は例を示しています。



4. [詳細（More Information）] をクリックします。
5. **証明書の表示**をクリックします。
6. 結果のダイアログボックスにタブページがある場合は、最上位 CA に対応するタブページをクリックします。
7. [その他（Miscellaneous）] セクションまでスクロールします。
8. [ダウンロード（Miscellaneous）] 行の [PEM（チェーン）（PEM (chain)）] をクリックします。次の図は例を示しています。



9. ファイルを保存します。
10. vCenter と Management Center の両方で、これらのタスクを繰り返します。

動的属性フィルタの作成

Cisco Secure 動的属性コネクタを使用して定義する動的属性フィルタは、アクセス コントロール ポリシーで使用できるダイナミックオブジェクトとして management center で公開されます。たとえば、財務部門の AWS サーバーへのアクセスを、Microsoft Active Directory で定義された財務グループのメンバーのみに制限できます。

（注） GitHub、Office 365、Azure サービスタグ では動的属性フィルタを作成できません。これらのタイプのクラウドオブジェクトは、独自の IP アドレスを提供します。

アクセス制御ルールの詳細については、動的属性フィルタを使用したアクセス制御ルールの作成 （52 ページ）を参照してください。

始める前に

コネクタの作成 （17 ページ）

ステップ 1 動的属性コネクタにログインします。

ステップ 2 [Dynamic Attributes Filters（ダイナミック属性フィルタ）] をクリックします。

- 新しいコネクタの追加：[追加（Add）]アイコン（＋）をクリックしてから、コネクタの名前をクリックします。
- コネクタの編集または削除：**その他**（⋮）をクリックしてから、行の末尾にある [編集（Edit）] または [削除（Delete）] をクリックします。

ステップ 3 次の情報を入力します。

項目	説明
名前	アクセス コントロール ポリシーおよび management center オブジェクトマネージャ（**[外部属性（External Attributes）]>[ダイナミックオブジェクト（Dynamic Object）]**）で動的フィルタを(ダイナミックオブジェクトとして) 識別するための一意の名前。
コネクタ	リストから、使用するコネクタの名前をクリックします。
クエリ	• 新しいフィルタの追加：[追加（Add）] アイコン（＋）をクリックします。 • フィルタの編集または削除：**その他**（⋮）をクリックしてから、行の末尾にある [編集（Edit）] または [削除（Delete）] をクリックします。

ステップ 4 クエリを追加または編集するには、次の情報を入力します。

項目	説明
キー（Key）	リストからキーをクリックします。キーはコネクタから取得されます。
操作	次のいずれかをクリックします。 • キーを値に正確に一致させるには、[等しい（Equals）]。

項目	説明
	• 値のいずれかの部分が一致する場合に、キーを値に一致させるには、[含む（Contains）]。
値	[任意（Any）] または [すべて（All）] をクリックし、リストから 1 つ以上の値をクリックします。[別の値を追加（Add another value）] をクリックして、クエリに値を追加します。

ステップ 5 [プレビューを表示（Show Preview）] をクリックして、クエリによって返されたネットワークまたは IP アドレスのリストを表示します。

ステップ 6 完了したら、[保存（Save）] をクリックします。

ステップ 7 （オプション）management center のダイナミックオブジェクトを確認します。

a) 最低限でもネットワーク管理者ロールを持つユーザとして management center にログインします。

b) **[オブジェクト（Objects）]** > **[オブジェクトマネージャ（Object Manager）]** をクリックします。

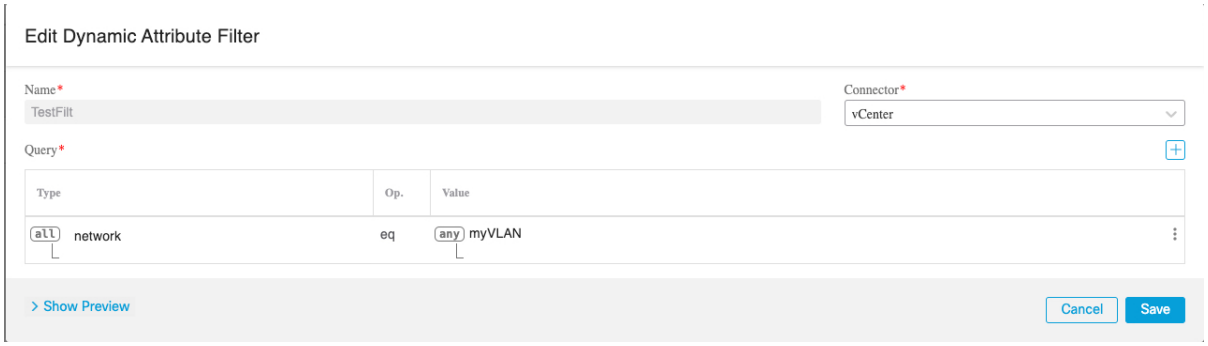
c) 左側のペインで、**[外部属性（External Attributes）]** > **[ダイナミックオブジェクト（Dynamic Object）]** をクリックします。
作成した動的属性クエリは、ダイナミックオブジェクトとして表示されます。

動的属性フィルタの例

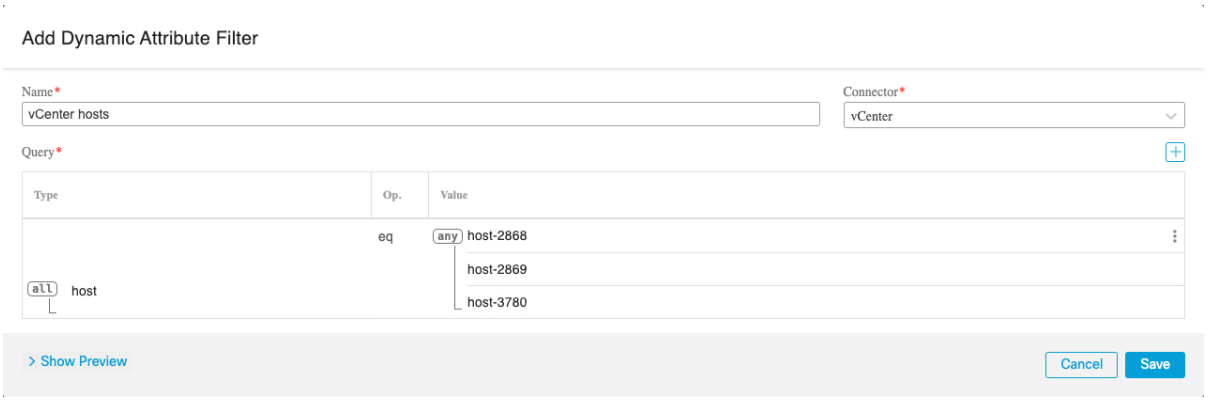
このトピックでは、動的属性フィルタの設定例をいくつか示します。

例：vCenter

次の例は、1 つの基準を示しています：VLAN。

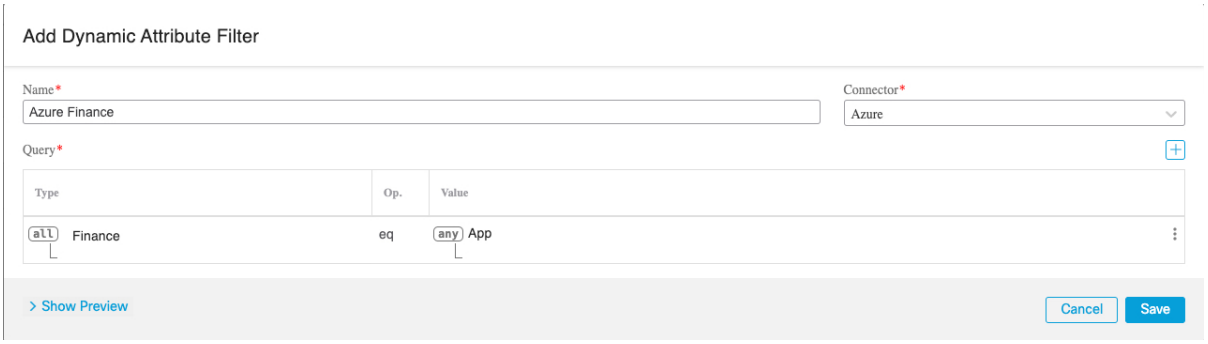


次の例は、OR で結合された 3 つの条件を示しています。クエリは 3 つのホストのいずれかに一致します。



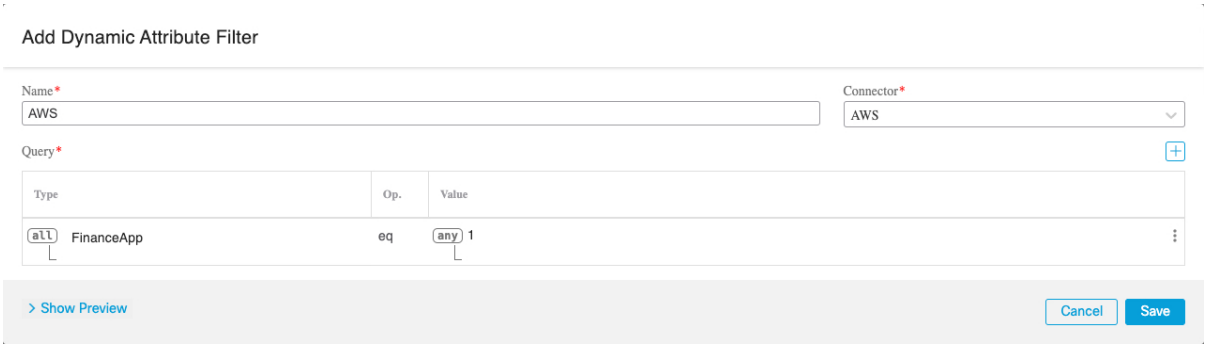
例：Azure

次の例は 1 つの条件を示しています：サーバーが財務アプリケーションとしてタグ付けされる。



例：AWS

次の例は、1 つの基準を示しています：値が 1 の FinanceApp。



認証局 (CA) チェーンの手動での取得

認証局チェーンを自動的に取得できない場合は、次のブラウザ固有の手順のいずれかを使用して、vCenter、NSX、またはManagement Centerに安全に接続するために使用される証明書チェーンを取得します。

証明書チェーンは、ルート証明書とすべての下位証明書です。

次に接続するには、これらの手順のいずれかを使用する必要があります。

- vCenter または NSX

  Azure または AWS に接続するために証明書チェーンを取得する必要はありません。

- Management Center

この手順を使用する前に、次の認証局チェーンの自動取得に関するセクションを参照してください。

- vCenter コネクタの作成 （30 ページ）

証明書チェーンの取得：Mac（Chrome および Firefox）

Mac OS で Chrome および Firefox ブラウザを使用して証明書チェーンを取得するには、この手順を使用します。

1. ターミナル ウィンドウを開きます。
2. 次のコマンドを入力します。

   ```
   security verify-cert -P url[:port]
   ```

   ここで、url は vCenter または Management Center への URL（スキームを含む）です。次に例を示します。

   ```
   security verify-cert -P https://myvcenter.example.com
   ```

   NAT または PAT を使用して vCenter または Management Center にアクセスする場合は、次のようにポートを追加できます。

   ```
   security verify-cert -P https://myvcenter.example.com:12345
   ```

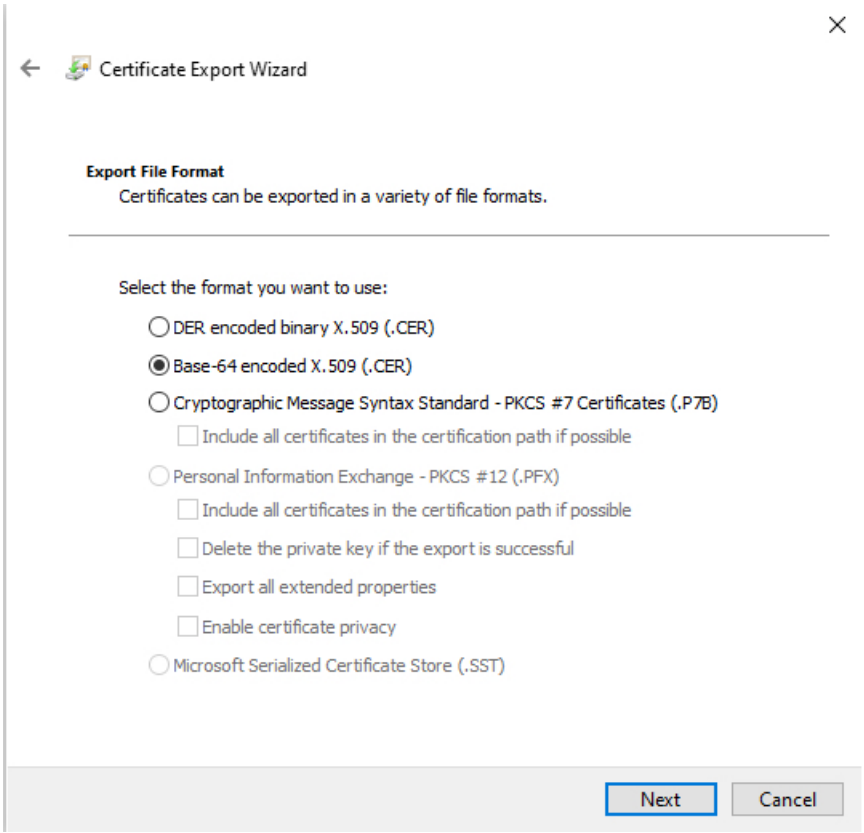
3. 証明書チェーン全体をプレーンテキストファイルに保存します。
   - すべての `-----BEGIN CERTIFICATE-----` および `-----END CERTIFICATE-----` 区切り文字を含めます。
   - 無関係なテキスト（たとえば、証明書の名前、山かっこ（<および>）に含まれるテキスト、および山かっこ自体を除外します。
4. vCenter と Management Center の両方で、これらのタスクを繰り返します。

証明書チェーンの取得：Windows Chrome

Windows で Chrome ブラウザを使用して証明書チェーンを取得するには、この手順を使用します。

1. vCenter または Chrome を使用してログインします。Management Center
2. ブラウザのアドレスバーで、ホスト名の左側にあるロックをクリックします。
3. [証明書（Certificate）] をクリックします。
4. [証明のパス（Certification Path）] タブをクリックします。
5. チェーンの最上位（つまり、最初）の証明書をクリックします。
6. **証明書の表示**をクリックします。
7. [詳細（Details）] タブをクリックします。
8. [ファイルにコピーする（Copy to File）] をクリックします。
9. プロンプトに従って、証明書チェーン全体を含むCER形式の証明書ファイルを作成します。

   エクスポートファイル形式の選択を求められたら、次の図に示すように、[Base 64 エンコード X.509（.CER）（Base 64-Encoded X.509 (.CER)）] をクリックします。



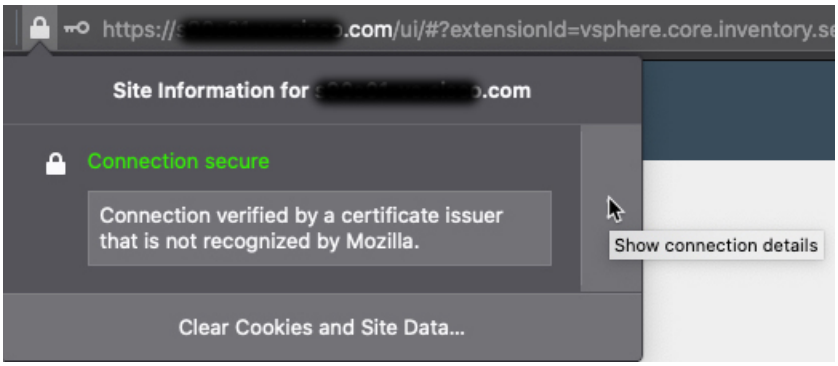
10. 指示に従ってエクスポートを完了します。
11. 証明書をテキストエディタで開きます。
12. チェーン内のすべての証明書に対してこのプロセスを繰り返します。

    テキストエディタに各証明書を最初から最後まで順番に貼り付ける必要があります。
13. vCenter と FMC の両方でこれらのタスクを繰り返します。

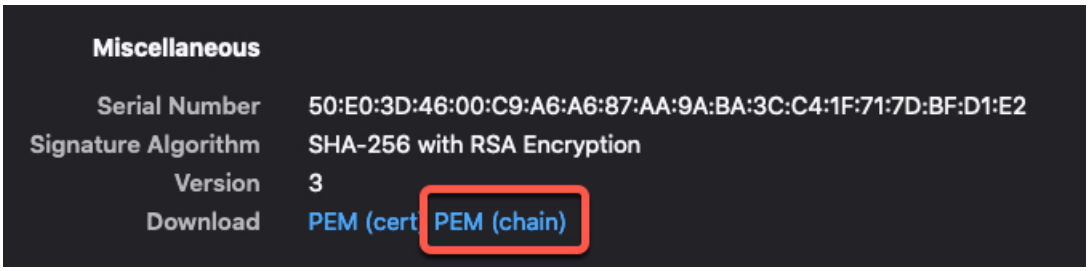
証明書チェーンの取得：Windows Firefox

Windows または Mac OS で Firefox ブラウザの証明書チェーンを取得するには、次の手順を使用します。

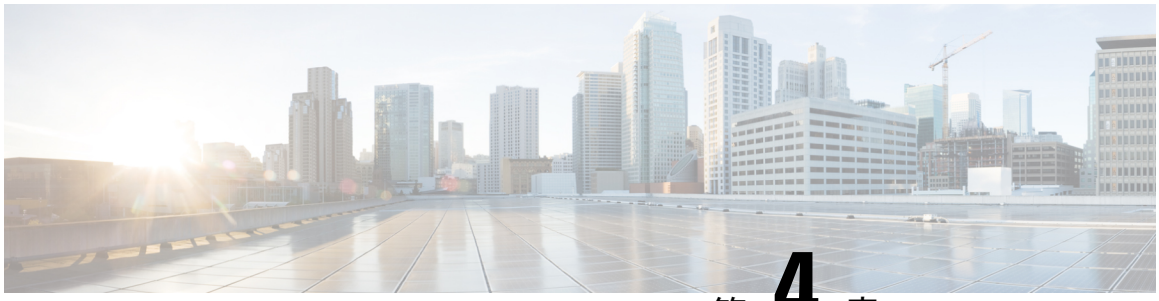
1. Firefox を使用して vCenter または Management Center にログインします。
2. ホスト名の左側にあるロックをクリックします。
3. 右矢印（[接続の詳細を表示（Show connection details）]）をクリックします。次の図は例を示しています。



4. [詳細（More Information）] をクリックします。
5. **証明書の表示**をクリックします。
6. 結果のダイアログボックスにタブページがある場合は、最上位 CA に対応するタブページをクリックします。
7. [その他（Miscellaneous）] セクションまでスクロールします。
8. [ダウンロード（Miscellaneous）] 行の [PEM（チェーン）（PEM (chain)）] をクリックします。次の図は例を示しています。



9. ファイルを保存します。
10. vCenter と Management Center の両方で、これらのタスクを繰り返します。



第 4 章

アクセス コントロール ポリシーでのダイナミックオブジェクトの使用

動的属性コネクタでは、アクセス制御ルールで、ダイナミックオブジェクトとしてmanagement centerに表示されるダイナミックフィルタを構成できます。

アクセス制御ルールのダイナミックオブジェクトについて

コネクタを作成し、動的属性フィルタを作成してそのコネクタに保存すると、ダイナミックオブジェクトが 動的属性コネクタから定義済み Cisco Secure Firewall に自動的にプッシュされます。

これらのダイナミックオブジェクトは、セキュリティグループタグ（SGT）の使用方法と同様に、アクセス制御ルールの [動的属性（Dynamic Attributes）] タブページで使用できます。送信元属性または接続先属性としてダイナミックオブジェクトを追加できます。たとえば、アクセス制御ブロックルールでは、ルール内の他の基準に一致するオブジェクトによって財務サーバーへのアクセスをブロックする接続先属性として財務ダイナミックオブジェクトを追加できます。

（注） GitHub、Office 365、Azure サービスタグ では動的属性フィルタを作成できません。これらのタイプのクラウドオブジェクトは、独自の IP アドレスを提供します。

動的属性フィルタを使用したアクセス制御ルールの作成

このトピックでは、ダイナミックオブジェクトを使用してアクセス制御ルールを作成する方法について説明します（これらのダイナミックオブジェクトは、前に作成した動的属性フィルタにちなんで命名されます）。

始める前に

動的属性フィルタの作成 （42 ページ）で説明されているように、動的属性フィルタを作成します。

（注） GitHub、Office 365、Azure サービスタグ では動的属性フィルタを作成できません。これらのタイプのクラウドオブジェクトは、独自の IP アドレスを提供します。

ステップ 1 management center にログインします。

ステップ 2 **[ポリシー（Policies）] > [アクセス制御（Access Control）]** をクリックします。

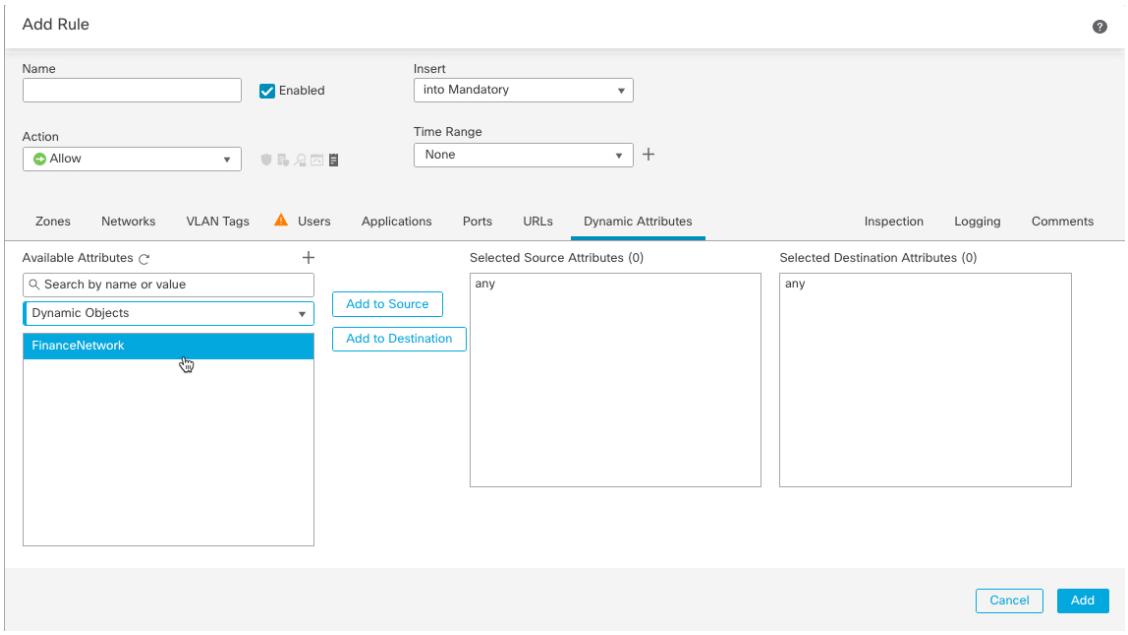
ステップ 3 アクセス コントロール ポリシーの横にある をクリックします。

ステップ 4 [ルールの追加（Add Rule）] をクリックします。

ステップ 5 [動的属性（Dynamic Attributes）] タブをクリックします。

ステップ 6 [使用可能な属性（Available Attributes）] セクションで、リストから [ダイナミックオブジェクト（Dynamic Objects）] をクリックします。

次の図は例を示しています。



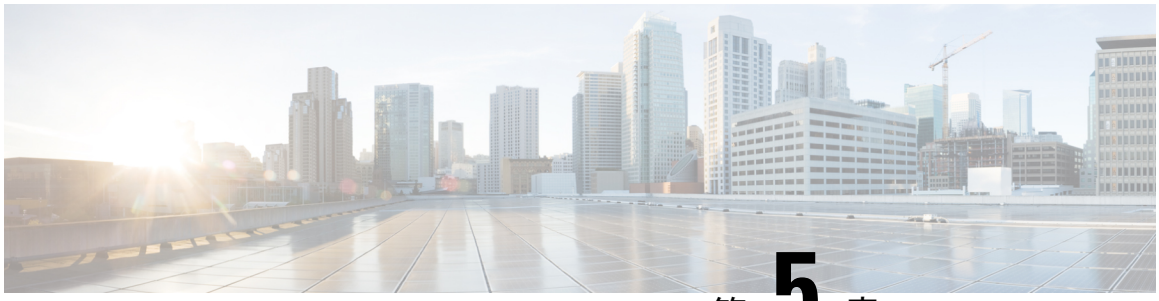
前の例は、Cisco Secure 動的属性コネクタ で作成された動的属性フィルタに対応する `FinanceNetwork` という名前のダイナミックオブジェクトを示しています。

ステップ 7 目的のオブジェクトを送信元または接続先属性に追加します。

ステップ 8 必要に応じて、ルールに他の条件を追加します。

次のタスク

『*Cisco Secure Firewall Management Center* デバイス構成ガイド』の「アクセス制御」の章（章へのリンク）



第 5 章

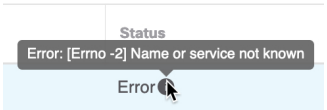
動的属性コネクタのトラブルシューティング

提供されているツールの使用など、動的属性コネクタで問題をトラブルシューティングする方法。

エラーメッセージのトラブルシューティング

問題：名前またはサービスが不明なエラー

このエラーは、コネクタのエラー状態にマウスを合わせると、ツールチップとして表示されます。次に例を示します。実際の表示とは異なる場合があります。



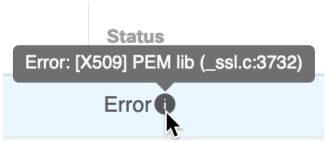
解決策：コネクタを編集して、次を確認します。

- ホスト名の末尾のスラッシュ
- （オンプレミス Firewall Management Center アダプタのみ。）ホスト名の先頭にあるスキーム（例：`https://`）
- パスワードが正しいことを確認する
- オンプレミス Firewall Management Center アダプタの場合、[FMCサーバー証明書（FMC Server Certificate）] フィールドの内容を確認します。

  詳細については、認証局 (CA) チェーンの手動での取得 （39 ページ）を参照してください。

問題：[X509 PEM lib]

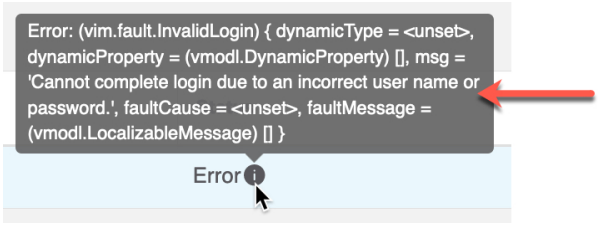
このエラーは、コネクタのエラー状態にマウスを合わせると、ツールチップとして表示されます。



解決策：コネクタを編集し、CAチェーンを確認します。詳細については、認証局(CA)チェーンの手動での取得 （39 ページ）を参照してください。

問題：正しくないユーザー名またはパスワード

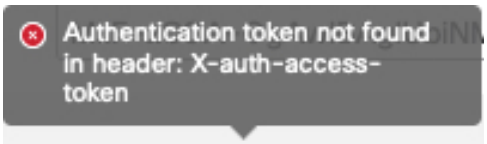
このエラーは、コネクタのエラー状態にマウスを合わせると、ツールチップとして表示されます。



解決策：コネクタを編集し、ユーザー名またはパスワードを変更します。

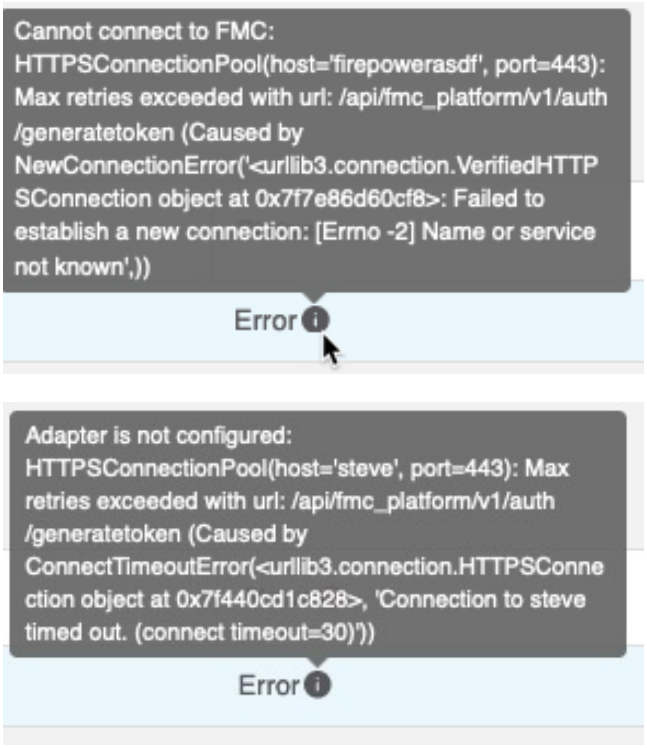
問題：ヘッダーに認証トークンが見つかりません

このエラーは、management centerに対する十分な権限を持たないアダプタユーザーとの接続をテストしようとすると表示されます。



問題：アダプタのタイムアウトまたは最大再試行エラー

このエラーは、アダプタのエラー状態にマウスを合わせると、ツールチップとして表示されます。



解決策：次のすべてを実行します。

- Management Center が実行されており、動的属性コネクタ からアクセスできることを確認します。
- [FMCサーバー証明書（FMC Server Certificate）] フィールドの内容を確認します。
- [IP] フィールドに入力した値が証明書の共通名と正確に一致していることを確認します。

詳細については、認証局 (CA) チェーンの手動での取得 （39 ページ）を参照してください。

コマンドラインを使用したトラブルシューティング

高度なトラブルシューティングと Cisco TAC との連携を支援するために、次のトラブルシューティング ツールを提供しています。これらのツールを使用するには、動的属性コネクタ が実行されている Ubuntu ホストに任意のユーザーとしてログインします。

コンテナステータスの確認

動的属性コネクタ Docker コンテナのステータスを確認するには、次のコマンドを入力します。

```
cd ~/csdac/app
sudo ./muster-cli status
```

出力例を次に示します。

```
============================== CORE SERVICES ==============================
      Name                     Command                  State    Ports
```

```
-------------------------------------------------------------------------------------------------
muster-bee     ./docker-entrypoint.sh run ...   Up             50049/tcp, 50050/tcp
 muster-etcd    etcd                             Up             2379/tcp, 2380/tcp

muster-ui      /docker-entrypoint.sh runs ...   Up (healthy)
0.0.0.0:443->8443/tcp,:::443->8443/tcp
muster-ui-backend   ./docker-entrypoint.sh run ...   Up             50031/tcp
============================ CONNECTORS AND ADAPTERS ============================
           Name                          Command               State     Ports
-------------------------------------------------------------------------------
muster-adapter-fmc.1         ./docker-entrypoint.sh run ...   Up      50070/tcp
muster-connector-vcenter.1   ./docker-entrypoint.sh run ...   Up      50070/tcp
```

動的属性コネクタ Docker コンテナの停止、起動、または再起動

./muster-cli status がコンテナが停止していることを示している場合、または問題が発生したときにコンテナを再起動するには、次のコマンドを入力できます。

停止と再起動：

```
cd ~/csdac/app
sudo ./muster-cli stop
sudo ./muster-cli start
```

起動のみ：

```
cd ~/csdac/app
sudo ./muster-cli start
```

アプリケーション デバッグ ロギングの有効化とトラブルシューティング ファイルの生成

Cisco TAC から推奨された場合は、デバッグロギングを有効にして、次のようにトラブルシューティング ファイルを生成します。

```
cd ~/csdac/app
sudo ./muster-cli debug-on
sudo ./muster-cli ts-gen
```

トラブルシューティング ファイル名は **ts-bundle-***timestamp***.tar** で、同じディレクトリに作成されます。

次の表は、トラブルシューティング ファイルとトラブルシューティング ファイル内のログの場所を示しています。

ロケーション	内容
/csdac/app/ts-bundle-*timestamp***/info**	`etcd` データベース格納ファイル
/csdac/app/ts-bundle-*timestamp***/logs**	コンテナログファイル
/csdac/app/ts-bundle-*timestamp***/status.log**	コンテナのステータス、バージョン、およびイメージのステータス

ダイナミックオブジェクトの確認

コネクタが management center でオブジェクトを作成していることを確認するには、management center で管理者として次のコマンドを使用します。

```
sudo tail f /var/opt/CSCOpx/MDC/log/operation/usmsharedsvcs.log
```

例：成功したオブジェクトの作成

```
26-Aug-2021 12:41:35.912,[INFO],(DefenseCenterServiceImpl.java:1442)
com.cisco.nm.vms.api.dc.DefenseCenterServiceImpl, ajp-nio-127.0.0.1-9009-exec-10
** REST Request [ CSM ]
** ID : 18b25356-fd6b-4cc4-8d27-bbccb52a6275
** URL: POST /audit
{
  "version": "7.1.0",
  "requestId": "18b25356-fd6b-4cc4-8d27-bbccb52a6275",
  "data": {
    "userName": "csdac-centos7",
    "subsystem": "API",
    "message": "POST
https://myfmc.example.com/api/fmc_config/v1/domain/e276abec-e0f2-11e3-8169-6d9ed49b625f
/object/dynamicobjects Created (201) - The request has been fulfilled and resulted in a
 new resource being created",
    "sourceIP": "192.0.2.103",
    "domainUuid": "e276abec-e0f2-11e3-8169-6d9ed49b625f",
    "time": "1629981695431"
  },
  "deleteList": []
}
```

認証局 (CA) チェーンの手動での取得

認証局チェーンを自動的に取得できない場合は、次のブラウザ固有の手順のいずれかを使用して、vCenter、NSX、またはManagement Centerに安全に接続するために使用される証明書チェーンを取得します。

証明書チェーンは、ルート証明書とすべての下位証明書です。

次に接続するには、これらの手順のいずれかを使用する必要があります。

- vCenter または NSX

  Azure または AWS に接続するために証明書チェーンを取得する必要はありません。

- Management Center

この手順を使用する前に、次の認証局チェーンの自動取得に関するセクションを参照してください。

- vCenter コネクタの作成 （30 ページ）

証明書チェーンの取得：Mac（Chrome および Firefox）

Mac OS で Chrome および Firefox ブラウザを使用して証明書チェーンを取得するには、この手順を使用します。

1. ターミナル ウィンドウを開きます。
2. 次のコマンドを入力します。

   ```
   security verify-cert -P url[:port]
   ```

ここで、url は vCenter または Management Center への URL（スキームを含む）です。次に例を示します。

```
security verify-cert -P https://myvcenter.example.com
```

NAT または PAT を使用して vCenter または Management Center にアクセスする場合は、次のようにポートを追加できます。

```
security verify-cert -P https://myvcenter.example.com:12345
```

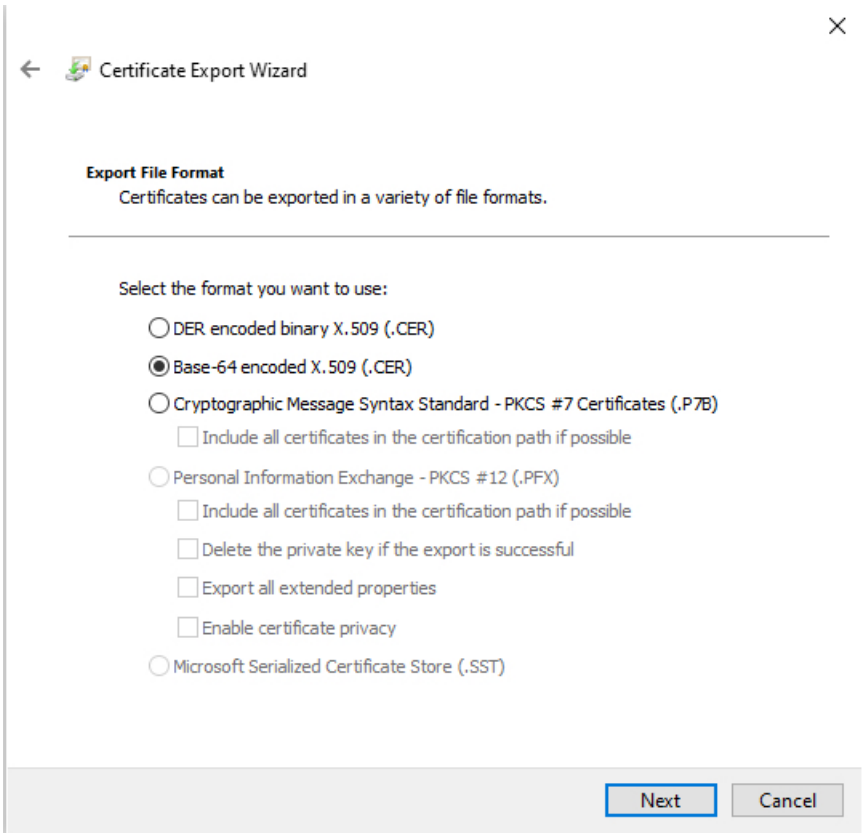
3. 証明書チェーン全体をプレーンテキストファイルに保存します。
   - すべての `-----BEGIN CERTIFICATE-----` および `-----END CERTIFICATE-----` 区切り文字を含めます。
   - 無関係なテキスト（たとえば、証明書の名前、山かっこ（<および>）に含まれるテキスト、および山かっこ自体を除外します。
4. vCenter と Management Center の両方で、これらのタスクを繰り返します。

証明書チェーンの取得：Windows Chrome

Windows で Chrome ブラウザを使用して証明書チェーンを取得するには、この手順を使用します。

1. vCenter または Chrome を使用してログインします。Management Center
2. ブラウザのアドレスバーで、ホスト名の左側にあるロックをクリックします。
3. [証明書（Certificate）] をクリックします。
4. [証明のパス（Certification Path）] タブをクリックします。
5. チェーンの最上位（つまり、最初）の証明書をクリックします。
6. **証明書の表示**をクリックします。
7. [詳細（Details）] タブをクリックします。
8. [ファイルにコピーする（Copy to File）] をクリックします。
9. プロンプトに従って、証明書チェーン全体を含むCER形式の証明書ファイルを作成します。

   エクスポートファイル形式の選択を求められたら、次の図に示すように、[Base 64 エンコード X.509（.CER）（Base 64-Encoded X.509 (.CER)）] をクリックします。



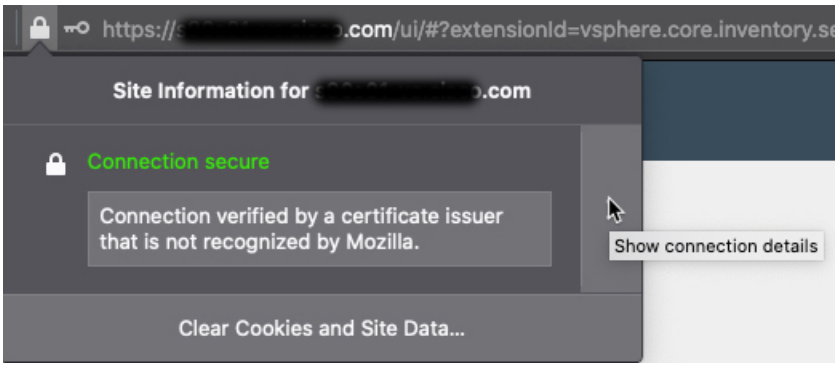
10. 指示に従ってエクスポートを完了します。
11. 証明書をテキストエディタで開きます。
12. チェーン内のすべての証明書に対してこのプロセスを繰り返します。

    テキストエディタに各証明書を最初から最後まで順番に貼り付ける必要があります。
13. vCenter と FMC の両方でこれらのタスクを繰り返します。

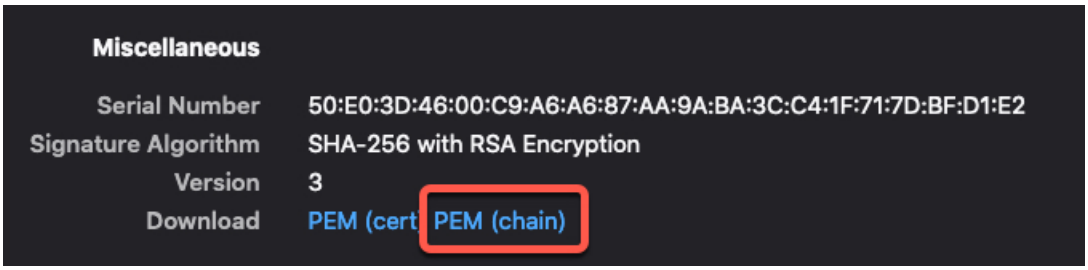
証明書チェーンの取得：Windows Firefox

Windows または Mac OS で Firefox ブラウザの証明書チェーンを取得するには、次の手順を使用します。

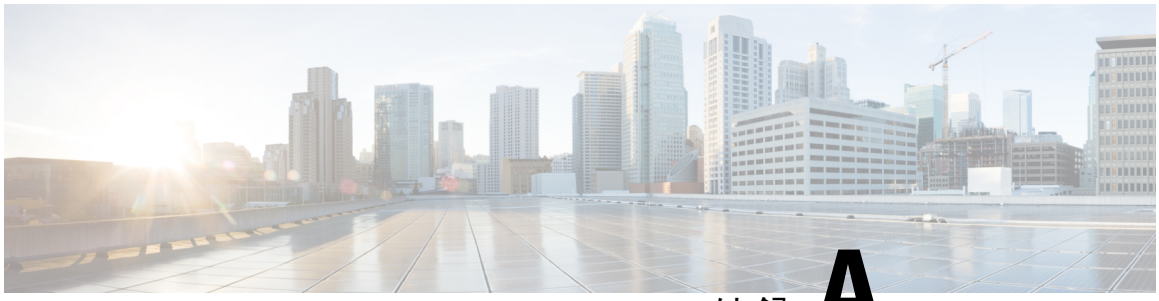
1. Firefox を使用して vCenter または Management Center にログインします。
2. ホスト名の左側にあるロックをクリックします。
3. 右矢印（[接続の詳細を表示（Show connection details）]）をクリックします。次の図は例を示しています。



4. [詳細（More Information）] をクリックします。
5. **証明書の表示**をクリックします。
6. 結果のダイアログボックスにタブページがある場合は、最上位 CA に対応するタブページをクリックします。
7. [その他（Miscellaneous）] セクションまでスクロールします。
8. [ダウンロード（Miscellaneous）] 行の [PEM（チェーン）（PEM (chain)）] をクリックします。次の図は例を示しています。



9. ファイルを保存します。
10. vCenter と Management Center の両方で、これらのタスクを繰り返します。



付録 A

セキュリティとインターネットアクセス

動的属性コネクタ がクラウド サービス プロバイダーおよび management center と通信するときに使用する URL のリスト。

セキュリティ要件

Cisco Secure 動的属性コネクタを保護するには、保護された内部ネットワークにそれをインストールしてください。動的属性コネクタは、使用可能なサービスとポートのうち必要なもののみを持つように設定されていますが、攻撃が到達できないように確保する必要があります。

動的属性コネクタ と management center が同じネットワーク上に存在している場合は、management center を 動的属性コネクタ と同じ保護された内部ネットワークに接続することができます。

アプライアンスの展開方法に関係なく、システム間通信は暗号化されます。それでも、分散型サービス拒否（DDoS）や中間者攻撃などの手段でアプライアンス間の通信が中断、ブロック、または改ざんされないよう何らかの対策を講じる必要があります。

インターネット アクセス要件

デフォルトでは、動的属性コネクタ は、ポート 443/tcp（HTTPS）で HTTPS を使用してインターネット経由で Firepower システムと通信するように構成されています。動的属性コネクタがインターネットに直接アクセスしないようにするために、プロキシサーバーを構成できます。

次の情報により、management center および外部サーバーとの通信に 動的属性コネクタ が使用する URL が通知されます。

セキュリティとインターネットアクセス

表 *3*：動的属性コネクタ *management center* アクセス要件

URL	理由
https://*fmc-ip***/api/fmc_platform/v1/auth/generatetoken**	認証
https://*fmc-ip***/api/fmc_config/v1/domain/***domain-id***/object/dynamicobjects**	GET および POST ダイナミックオブジェクト
https://*fmc-ip***/api/fmc_config/ v1/domain/***domain-id***/object/dynamicobjects/***object-id***/mappings?action=add**	マッピングを追加します
https://*fmc-ip***/api/fmc_config/v1/domain/***domain-id* **/object/dynamicobjects/***object-id***/mappings?action=remove**	マッピングを削除します

表 *4*：動的属性コネクタ *vCenter* アクセス要件

URL	理由
https://*vcenter-ip***/rest/com/vmware/cis/session**	認証
https://*vcenter-ip***/rest/vcenter/vm**	VM 情報を取得します
https://*nsx-ip***/api/v1/fabric/virtual-machines/***vm-id*	仮想マシンに関連付けられた NSX-T タグを取得します

DockerHub から Amazon ECR への移行

Cisco Secure 動的属性コネクタ の Docker イメージは、Docker Hub から Amazon Elastic Container Registry（Amazon ECR）に移行されています。

新しいフィールドパッケージを使用するには、ファイアウォールまたはプロキシから次のすべての URL へのアクセスを許可する必要があります。

- https://public.ecr.aws

  個々のフィールドパッケージをダウンロードするには、Amazon ECR ギャラリーで**muster** を検索します。

- https://csdac-cosign.s3.us-west-1.amazonaws.com

動的属性コネクタ Azure のアクセス要件

動的属性コネクタは、組み込みの SDK メソッドを呼び出してインスタンス情報を取得します。これらのメソッドは、https://login.microsoft.com（認証用）と https://management.azure.com（インスタンス情報の取得用）を内部的に呼び出します。

翻訳について
このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。